DISSERTATION

Eine datenbankbasierte Entwicklungsumgebung zur Online-Generierung von multimedialen Lernmodulen für das Lernen im Fach Technik

zur Erlangung des akademischen Grades
eines Doktor-Ingenieurs (Dr.-Ing.)

des Fachbereichs Maschinenwesen
der Universität Duisburg-Essen
am Campus Essen

vorgelegt von
Jürgen Wehling
aus Gelsenkirchen

Datum der mündlichen Prüfung:	7. Januar 2004
1. Gutachter:	Prof. Dr. rer. nat. W. Haupt
2. Gutachter:	Prof. Dr.-Ing. E. Sauer

Vorwort

Die vorliegende Arbeit entstand in der Zeit von Januar 2001 bis Juli 2003 unter Betreuung der beiden Professoren des Faches Technologie und Didaktik der Technik TUD.

Herrn Prof. Dr. rer. nat. W. Haupt gebührt mein besonderer Dank für seine kompromisslose Bereitschaft, inhaltliche und fachliche Fragestellungen dieser Arbeit jederzeit zu diskutieren.

Herrn Prof. Dr.-Ing. E. Sauer danke ich sehr herzlich für die intensive Förderung dieser Arbeit und für die bereitwillige Übernahme des Korreferats.

Zudem darf ich mich besonders intensiv bei meiner Frau Christiane bedanken, die zu jeder Zeit sehr viel Verständnis für meine Arbeit entgegen brachte sowie bei meiner im August 2003 geborenen Tochter Helena, die mich ab vier Uhr morgens regelmäßig an den beginnenden Tag erinnerte.

Gelsenkirchen, im Januar 2004 — Jürgen Wehling

Form der vorliegenden Arbeit

Die gesamte Arbeit besteht aus dem vorliegenden **schriftlichen Teil** und einem in voller Form nur über das Internet zugänglichen **online-basierten Teil**[1] als **PDF-Version**, abrufbar über den Universitätsserver des HRZ unter der URI[2] http://www.uni-essen.de/~kte010/diss/DBIDE.pdf, die alle internen und ins Internet führenden Verweise in Form von Hyperlinks ausweist.

Im **schriftlichen Teil** wird eine begründete Umsetzung der Forderungen für ein Lernen mit multimedialen Inhalten dargestellt und gezeigt, wie sich diese Forderungen realisieren lassen.

Der **online-basierte Teil** als **PDF-Version** stellt durch die voll funktionsfähige *datenbankbasierte Entwicklungsumgebung* das in der Praxis überprüfbare Ergebnis dieser Arbeit in Form einer **Internet-Präsenz** dar: er konkretisiert durch das dynamische Zusammenspiel von komplexen Programmen die im schriftlichen Teil formulierten Forderungen und verfügt durch die Implementierung von Textmarken und Hyperlinks über eine eigene Verweisstruktur. Diese liefert neben einer textorientierten Referenzierung zusätzlich die Möglichkeit, internetbasierte Quellen zu referenzieren[3]. So sind neben der **Internet-Präsenz** (MMDB-TU, INTEGER) der *datenbankbasierten Entwicklungsumgebung* zusätzlich der selbst erstellte **Programmcode** (HTML, XHTML, PHP, Perl), das **Benutzer-Handbuch** (Video-Sequenzen) sowie die **Server-Einrichtung** (L.A.M.P.-Ansatz) ausschließlich durch Hyperlinks erreichbar. Dieser Aufbau wurde im Hinblick auf eine überschaubare Struktur gewählt.

Diese Vorgehensweise ermöglicht bei einer bestehenden Internet-Verbindung den direkten Zugriff auf die referenzierenden Elemente dieser Arbeit:

- **Internet-Präsenz:** Die Ergebnisse dieser Arbeit sollen global zugänglich sein. Zu diesem Zweck wurden zwei URI[4] gleichen Inhalts aufgesetzt, die über die Hauptsuchmaschinen des Internet in adäquater Form indexiert werden können, wobei sie identische Zugänge zum Hauptrechner anbieten: http://www.mmdb-tu.de und http://www.integer-tu.de.
- **Programmcode:** Der kommentierte Programmcode, der in ausgedruckter Form mehr als 1000 DIN A4 Seiten umfasst, lässt sich über den URI http://www.mmdb-tu.de/programmcode/quellcodes.htm abrufen und einsehen. Sämtliche PHP-, Perl-, Java und JavaScript-Programme sind unter dieser Adresse dokumentiert.

---

[1] Ein Offline-Zugang ist nicht möglich, da die meisten Programme in PHP und Perl geschrieben sind und ausschließlich serverseitig ausgeführt werden.

[2] URI ist der technisch korrekte Begriff des URL und bezeichnet eine eindeutig identifizierte Adresse im Internet.

[3] Sämtliche Internet-basierten Quellen liegen zusätzlich als sog. **long lasting links** (**lll**) vor, d. h. sie sind dauerhaft archiviert und daher in jedem Fall referenzierbar.

[4] Dieser Schritt ist notwendig, um die *datenbankbasierte Entwicklungsumgebung* mit zwei fest vorgegebenen Suchbegriffen ('MMDB-TU' und 'INTEGER') bei den Hauptsuchmaschinen zu indexieren und so einer breiteren Öffentlichkeit zugänglich zu machen.

- **Benutzer-Handbuch:** Auf die textbasierte Erstellung eines Benutzer-Handbuchs wurde zugunsten einer ansprechenderen Dokumentationsform verzichtet: es kommen Video-Sequenzen der wichtigsten Situationen beim Arbeiten mit der MMDB-TU und INTEGER zum Einsatz. Z. Zt. stehen diese unter der URI http://www.mmdb-tu.de/benutzerhandbuch/mmdb-tu_videos.htm bzw. http://www.mmdb-tu.de/benutzerhandbuch/integer-tu_videos.htm bereit. Hier finden sich auch Beispiele für einen DC-konformen Aufbau von Objekten und Modulen. Als Alternative zum Download der Video-Sequenzen wird zum Jahresende eine menügeführten DVD zur Verfügung stehen, die über den Autor dieser Arbeit angefordert werden kann.
- **Server-Einrichtung:** Unter dem URI http://www.mmdb-tu.de/servereinrichtung/lamp.htm ist die komplette Einrichtung eines Webservers und Datenbankservers nach dem L.A.M.P.-Ansatz beschrieben.

Kurzfassung

Eine Bestandsaufnahme multimedialen Lernens unter Einsatz der Neuen Medien zeigt, dass hohe Anforderungen an die Gestaltung von digitalisierten Lehr- und Lernmaterialien gestellt werden, wobei sich sowohl der Lernende als auch der Lehrende der Herausforderung einer permanenten Erweiterung seiner eigenen medialen Kompetenz stellen muss. Bedingt durch eine neue Studienordnung, eine neue Lehramtsprüfungsordnung und neue Richtlinien für die gymnasiale Oberstufe mit Schwerpunkten im informationstechnischen Bereich ist das Fach Technik in der Pflicht, eine mediale Kompetenzerweiterung auf der Basis einer informationstechnischen Grundbildung anzubieten (Kapitel 2).

Über drei wesentliche, aufeinander folgende Schritte lässt sich multimediales Lernen zu einer ganzheitlichen Konzeption zusammenführen (Kapitel 3.1):

1. Die standardisierte Entwicklung von Objekten (Content)
2. Das strukturierte Ablegen und neu strukturierte Kombinieren von Objekten zu Lernobjekten (Lernmodule, Lerneinheiten, Lehrgänge)
3. Der Einsatz strukturierter Lernobjekte in Lernplattformen.

Der o.g. Schritt 2 umschreibt die beiden zentralen Aspekte dieser Arbeit. Ein strukturiertes Ablegen von multimedial orientierten Objekten (Content) erfolgt auf der Grundlage eines relationalen Datenbankmanagement-Systems (RDBMS). Ein geeignet gewähltes RDBMS stellt die Basis für eine Entwicklungsumgebung dar, um Objekte zu Lernobjekten (Lernmodule, Lerneinheiten und Lehrgänge) neu strukturiert zu kombinieren. Diese *datenbankbasierte Entwicklungsumgebung* schließlich liefert einen realisierbaren Ansatz, um Lernobjekte für einen Einsatz in Lernplattformen online vorzubereiten und anschließend ebenfalls online zur Verfügung zu stellen.

Sowohl das RBDMS, als auch die darauf aufsetzende Entwicklungsumgebung lassen sich durch einen L.A.M.P.-Ansatz realisieren. Dieser Ansatz basiert konsequent auf Open-Source-Produkten, die als freie Software weltweit kostenlos verfügbar sind. Damit ist sichergestellt, dass jeder Nutzer unabhängig vom jeweils verwendeten Betriebssystem mit der *datenbankbasierten Entwicklungsumgebung* arbeiten kann. Auf der Grundlage einer so implementierten Schnittstelle bietet sich die Möglichkeit, digitalisierte Inhalte in den wichtigsten, netzgestützten Formaten allen potentiellen Anwendern zur Verfügung zu stellen; lediglich eine informationstechnische Grundbildung ohne spezielle zusätzliche Kenntnisse ist die clientseitige Voraussetzung für deren Nutzung (Kapitel 3.2).

Realisiert wird diese Nutzerschnittstelle durch einen modularen Ansatz, wobei die *datenbankbasierte Entwicklungsumgebung* eine komfortable Möglichkeit bietet, neue Lernobjekte, die systematisch mit Hilfe von Meta-Daten klassifiziert wurden, unter neuen Schwerpunktsetzungen strukturiert zu erzeugen. Hierbei finden die folgenden Definitionen Verwendung:

- Ein *Modul* besteht aus wenigstens zwei Objekten, die zu einer thematisch zusammenhängenden Einheit gefügt sind, während ein *Objekt* eine kleinste, nicht mehr weiter unterteilbare, zusammenhängende Einheit (z. B. ein Bild, einen thematisch geschlossenen Text, ein Applet, eine Animation, etc.) darstellt. Module (Lernobjekte) liegen im XHTML-Format vor.
- Objekte/Module werden grundsätzlich durch Meta-Angaben beschrieben. Die Grundlage für eine klassifizierende Beschreibung ist die DCMI.

Dieser o.g. modulare Ansatz ist in verschiedenen Projekten im Fach Technologie und Didaktik der Technik im Rahmen der Techniklehrerausbildung entstanden. Hierbei ging es im wesentlichen um eine Entwicklung von Lernmodulen für das Fach (Kapitel 4.1).

Ein Ergebnis dieses im Fach Technik konsequent umgesetzten modularen Ansatzes ist die multimedial ausgerichtete Datenbank *mmdb*, welche durch die zwei graphisch orientierten Nutzerschnittstellen MMDB-TU (MultiMediaDatenBank-TechnikUnterricht) und INTEGER (Integrierte Entwicklungsumgebung für die Generierung von Lernobjekten) speziell auf die Erfordernisse dieses Faches zugeschnitten ist. Beide Nutzerschnittstellen, MMDB-TU und INTEGER, stellen zusammen mit der Datenbank *mmdb* die *datenbankbasierte Entwicklungsumgebung* dar.

Auf der Grundlage einfacher Objekte, die in der Datenbank *mmdb* in strukturierter Form vorliegen, lassen sich über INTEGER Lernobjekte mit neuen Schwerpunktsetzungen generieren und über die MMDB-TU wieder zurückschreiben. Dabei können neue Lernobjekte neben technikspezifischen Erweiterungen auch lernzielrelevante Kriterien aufweisen (Kapitel 4.2).

Mit INTEGER und der MMDB-TU steht ein System zur Verfügung, das

- die Lücke zwischen bloßem Content und e-Learning füllt,
- eine Generierung fachspezifischer Lernobjekte ermöglicht und
- Lernobjekte fachdidaktisch für einen Einsatz in Lern- und Kommunikationsplattformen ergänzt.

Die voll funktionsfähige *datenbankbasierte Entwicklungsumgebung* ist erreichbar über die URI http://www.mmdb-tu.de und http://www.integer-tu.de, und stellt die überprüfbare Realisierung der in dieser Arbeit aufgestellten Forderungen dar. Spezielle Beispiele und Screenshots der *datenbankbasierten Entwicklungsumgebung* finden sich im Anhang.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

Abkürzungsverzeichnis

Deutsche Erklärungen der in dieser Arbeit verwendeten Abkürzungen finden sich im Internet-Glossar [Didszuns, 2003].

ADL:	Advanced Distributed Learning Initiative
AICC:	Aviation Industry Computer Based Training Commitee
ANSI:	American National Standards Institute
ARIADNE:	Alliance of Remote Instructional Authoring and Distribution Networks for Europe
ASP:	Active Server Pages
ATA:	Advanced Technology Attachments
AVMZ:	Audiovisuelles Medienzentrum
BLBS:	Bundesverband der Lehrerinnen und Lehrer an beruflichen Schulen
bm:bwk:	Bundesministerium für Bildung, Wissenschaft und Kultur, Wien, Österreich
BMBF:	Bundesministerium für Bildung und Forschung
CBT:	Computer Based Training
CGI:	Common Gateway Interface
CORBA:	Common Request Broker Architecture
CPAN:	Comprehensive Perl Archive Network
CRT:	Cathode Ray Tube
CSS:	Cascading Stylesheets
DAAD:	Deutscher akademischer Austauschdienst
Daemon:	Disk and execution monitor
DBI:	Database Interface
DBMS:	Datenbank Management System
DC:	Dublin Core
DCMI:	Dublin Core Metadata Initiative
DCQ:	Dublin Core Qualifiers
DGTB:	Deutsche Gesellschaft für Technische Bildung
DHTML:	Dynamic Hypertext Markup Language
DTD:	Document Type Definition
EBNF:	Extended Backus-Naur Form
EGTB:	Europäische Gesellschaft für Technische Bildung
EJB:	Enterprise JavaBeans
ER:	Entity Relationship
FTP:	File Transfer Protocoll

GEM:	Gateway to Educational Materials
GNU GPL:	GNU Is Not UNIX General Public License, Free Software Foundation, Inc.
GUI:	Graphical User Interface
HIS:	Hochschul-Informations-System GmbH
HRZ:	Hochschul-Rechenzentrum
HTML:	Hypertext Markup Language
IDE:	Integrated Drive Electronics
IDLE:	Integrated Distributed Learning Environment
IEEE:	Institute of Electrical and Electronic Engineers, USA
IETF:	Internet Engineering Taskforce
IFPL:	Institut für Fachdidaktik Physik und Lehrerbildung
IIS:	Internet Information Server
ILIAS:	Integriertes Lern-, Informations- und Arbeitskooperationssystems
IMS:	Instructional Managing System
IMT:	Internet Media Type
INTEGER:	Integrierte Entwicklungsumgebung für eine Generierung von Lernobjekten
ISO:	International Standards Organisation
ISP:	Internet Service Provider
L.A.M.P.:	Linux, Apache, MySQL, PHP/Perl
LAN:	Local Area Network
LCD:	Liquid Cristal Display
LCSH:	Library of Congress Subject Headings
LIT:	Leipziger Informatiktage
LMS:	Learning Management System
LOM:	Learning Objects Metadata
LPF:	Lernplattform
LTSC:	Learning Technology Standards Committee
MILESS:	Multimedialer Lehr- und Lernserver Essen
MIME:	Multipurpose Internet Mail Extensions
MMDB-TU:	Mutlimedia Datenbank Technikunterricht
MPEG:	Motion Picture Expert Group
MSJK:	Ministeriums für Schule, Jugend und Kinder des Landes Nordrhein-Westfalen
MSWF:	Ministerium für Schule, Wissenschaft und Forschung des Landes Nordrhein-Westfalen

MWF:	Ministeriums für Wissenschaft und Forschung des Landes Nordrhein-Westfalen
OASIS:	Organization for the Advancement of Structured Information Standards
OpenUSS:	Open University Support System
OSI:	Open Systems Interconnection
PATT:	Pupils' Attitude Towards Technology
PDF:	Portable Document Format
Perl:	Practical Extraction and Report Language
PHP:	PHP Hypertext Präprozessor
PICS:	Platform for Internet Content Selection
RAID:	Redundant Array of Independent Disks
RDBMS:	Relationales Datenbank Management System
RDF:	Resource Description Framework
RFC:	Request for Comments
SCO:	Santa Cruz Operation Inc.
SCORM:	Sharable Content Object Reference Model
SGML:	Standard Generalized Markup Language
SQL:	Structured Query Language
SPL:	Stored Procedure Language
StO:	Studienordnung
SuSE:	Software- und System-Entwicklung mbH, Deutschland
TDT:	Technologie und Didaktik der Technik an der Universität Duisburg Essen, Campus Duisburg
TERC:	Technology Education Research Conference
TFT:	Thin Film Transistor
T.I.M.E:	Telekommunikation, Internet, Multimedia, Elektronik
T-SQL:	Transact-Structured Query Language
TUD:	Technologie und Didaktik der Technik an der Universität Duisburg Essen, Campus Essen
TUF:	Technik Unterricht Forum, Verband der Techniklehrer für die gymnasiale Oberstufe in NRW, Deutschland
URI:	Uniform Resource Identifier
URL:	Uniform Resource Locator
UrhG-E:	Urhebergesetz unter Berücksichtigung der Urheberrechtsrichtlinie der Europäischen Union
USV:	Unterbrechungsfreie Stromversorgung
UVM:	Universitätsverbund Multimedia

VDI:	Verein Deutscher Ingenieure, Deutschland
VIB:	Virtualisierung im Bildungsbereich
VLE:	Virtual Learning Environment
W3:	World Wide Web
W3C:	World Wide Web Consortium
W3C-DTF:	World Wide Web Consortium Date and Time Formats
W.A.M.P.:	Windows, Apache, MySQL, PHP/Perl
WAN:	Wide Area Network
WBT:	Web Based Training
WWW:	World Wide Web
WYSIWYG:	What You See Is What You Get
XHTML:	Extensible Hypertext Markup Language
XML:	Extensible Markup Language
XSL:	Extensible Stylesheet Language

1 Einleitung und Zielsetzung

Die hier vorliegende Arbeit bezieht sich auf ein vom Bundesministerium für Bildung und Forschung gefördertes Projekt.

In Zusammenarbeit der Universität Duisburg-Essen, Campus Essen mit der Technischen Universität Braunschweig werden durch eine Kooperative des Faches Technologie und Didaktik der Technik (Essen) mit dem Institut für Allgemeine Technikpädagogik (Braunschweig) "Lehr- und Lernmodule im Baukastenmodus zu den disziplinübergreifenden Bereichen Stoff-, Energie- und Informationsumsetzende Systeme für die Studiengänge Lehrämter Technik Sekundarstufe I und Sekundarstufe II" unter dem Rahmenthema "Versorgung eines Wohnhauses mit Wärme und elektrischer Energie".

Die zu erstellenden Lehr- und Lernmodule sollten, um dem Baukastenprinzip zu genügen, in einer systematisch geordneten Form vorliegen. Es bot sich daher an, diese Module durch eine interaktive, multimedial orientierte Datenbank zu systematisieren, um sie für eine spätere Verwendung gezielt abrufen zu können. Zu diesem Zweck schuf der Autor dieser Arbeit eine "Integrierte Entwicklungsumgebung für eine Online-Generierung von multimedial ausgerichteten Lernobjekten", wobei neben fachwissenschaftlichen auch fachdidaktische Aspekte des Faches Technologie und Didaktik der Technik (im folgenden *Technik* genannt) besondere Berücksichtigung finden. Das Arbeiten mit dieser integrierten Entwicklungsumgebung erfordert clientseitig keine speziellen informationstechnischen Kenntnisse.

Die vorliegende Arbeit beschreibt den Weg, der von der Ausgangslage über die Problematisierung zur Realisierung der im Thema genannten *datenbankbasierten Entwicklungsumgebung* führt. Er gibt an wesentlichen Stationen den für diese Arbeit wichtigen Teil der zugrunde liegenden ganzheitlichen Konzeption des Lernens mit multimedialen Inhalten wieder und schließt als Resümee mit einer reflektierenden Betrachtung durch das Kapitel Zusammenfassung und Ausblick.

Die in Kapitel 2 zur Ausgangslage dargelegten Ausführungen zeichnen die Entwicklung und den Stellenwert des Faches Technik bis zum jetzigen Zeitpunkt nach. Hier wird verdeutlicht, dass sich dieses Fach als ein dynamisches Fach sowohl fachverbindenden Ingenieurdisziplinen, als auch der Herausforderung, die durch das Aufkommen der Neuen Medien entsteht, jeder Zeit zu stellen hat. Das bedeutet: **Erweiterung der medialen Kompetenzen des Einzelnen**. Förderinitiativen des Bundes und der Länder bieten hier wichtige Ansatzpunkte, die vom Fach aufgegriffen und mit zunehmender medialer Kompetenz seit Jahren regelmäßig umgesetzt werden. Durch eine Forcierung bildungspolitisch motivierter Konzepte auf den Gebieten des e-Learning werden die Hochschulen zunehmend in einen globalen Bildungsmarkt gedrängt, dessen Anforderungen sie erfüllen müssen, um weiter zu bestehen.

Der erste Teil von Kapitel 3 greift die Position des Faches Technik im fachdidaktischen Bereich auf und problematisiert über ein dreistufiges Modell eines fachbezogenen multimedialen Lernens die Notwendigkeit der Existenz eines speziellen RDBMS. Angesiedelt zwischen unstrukturiertem Content und komplexen Plattformen zum e-Learning wird hier dargestellt, in welcher Form ein didaktisch begründetes multimediales Lernen unter strukturiertem Einsatz der Neuen Medien vorbereitet werden kann.

Im zweiten Teil von Kapitel 3 wird der Ansatz nach L.A.M.P. begründet, der schließlich softwareseitig die Basis für ein multimediales Lernen im Fach liefert. Ausgehend von einfachen Objekten wird an dieser Stelle der gewählte modulare Ansatz begründet. Er

bietet über die konkrete Realisierungsmöglichkeit eines Webservers die Grundlage für die Implementierung einer *datenbankbasierten Entwicklungsumgebung*. Die zu erwartenden Umsetzungsergebnisse werden im anschließenden Kapitel konkretisiert.

Das Kapitel 4 ist zweigeteilt und enthält neben der Umsetzung des modularen Ansatzes die Realisierung der zugehörigen Datenbank und deren Nutzerschnittstellen mit Entwicklungsumgebung. Im Zusammenhang mit dem modularen Ansatz wird hier noch einmal auf die Notwendigkeit strukturierter Beschreibungssätze eingegangen. Neben den Entscheidungsbegründungen für die Realisierung der beiden Nutzerschnittstellen MMDB-TU (Multimediadatenbank Technikunterricht) und INTEGER (Integrierte Entwicklungsumgebung für eine Generierung von Lernobjekten) wird durch exemplarisch ausgewählte Code-Segmente, Struktogramme und Flussdiagramme der realisierte Weg dargestellt.

Aufrufbar durch eine javabasierte Menüstruktur arbeiten mehr als 200 browserunabhängig erstellte Webseiten mit integrierten Javascript-Programmen, sowie ca. 60 Perl- bzw. PHP-basierte Programme von zum Teil beträchtlichem Umfang zusammen. Es ergeben sich mehr als 1000 DIN A4 Seiten selbst erstellten Programmcodes, dessen Darstellung in ausgedruckter Form unangebracht erscheint. Der kommentierte Programmcode lässt sich über den URI http://www.mmdb-tu.de/programmcode/quellcodes.htm abrufen und einsehen.

Abschließend fasst Kapitel 5 in Form eines Resümees noch einmal wesentliche Aspekte dieser Arbeit in reflektierender Form zusammen und liefert einen kurzen Ausblick auf die künftige Weiterführung und den weiteren Ausbau der *datenbankbasierten Entwicklungsumgebung*. Dabei wird neben fachwissenschaftlichen und fachdidaktischen Aspekten auch die besondere Stellung des Faches Technik im Umfeld von Schule und Hochschule und seine mögliche weitere Entwicklung angesprochen.

2 Ausgangslage

Nach Einführung der Neigungsdifferenzierung und der Hinwendung zu stärker praxisbezogenen Lerninhalten an allgemein bildenden Schulen wurden 1973 in Nordrhein-Westfalen an den am 1. August 1972 gegründeten Hochschul-Standorten Essen und Duisburg das Fach "Technologie und Didaktik der Technik" [TUD, 2003; TDT, 2001] eingerichtet, welches die Grundlage für die Einführung des Technikunterrichts in den verschiedenen Schulformen und Schulstufen schafft und die darauf bezogene Lehrerausbildung aufnehmen sollte[1].

In Nordrhein-Westfalen bezieht sich die Lehrerausbildung auf die Sekundarstufen II, I und die Primarstufe[2]; an der Hochschule Duisburg-Essen[3] werden Lehrer für die Sekundarstufen I und II im Fach Technik ausgebildet.

Auf der Grundlage eines 1975 im Auftrag des Kultusministeriums NRW entwickelten "Curriculum Technik" [Bader, 1975] und dem 1976 veröffentlichten "Ansatz einer Didaktik der Technik für den Technikunterricht der Sekundarstufe II" [Haupt, 1976] entstanden die ersten Richtlinien[4] für die gymnasiale Oberstufe des Faches Technik [Richtlinien, 1981]. In diesem Zusammenhang sei auf die systemorientierte Strukturierung des Faches Technik durch ein kybernetisch-systemtheoretisches Modell hingewiesen, das im wesentlichen durch die Kategorien Stoff-, Energie- und Informationsumsatz beschrieben wird.

Insbesondere die für das Fach Technik wichtigen ingenieurwissenschaftlichen Disziplinen sind in sehr differenzierter Vielfalt (es existieren fast 200 Disziplinen [BLBS, 2003]) vorhanden. Eine technikspezifische Klassifikation technischer Sachsysteme ordnet z. B. dem Energieumsatz die Systeme "Thermisches Kraftwerk", "Fernwärmeverbund" und "Nachtspeicherheizung" mit den entsprechenden Funktionen "Wandlung", "Transport" und "Speicherung" zu [Ropohl, 1975]. Auf diesem Wege werden die vielfältigen Inhalte der großen Anzahl ingenieurwissenschaftlicher Disziplinen strukturiert.

Zum Selbstverständnis des Faches gehören

- der Bezug zu technikwissenschaftlichen Grundlagen[5] (u.a. manifestiert durch einen in den Richtlinien zur Sekundarstufe II des Faches formulierten wissenschaftspropädeutischen Ansatz),

---

[1] Seit dem 1. Januar 2003 gibt es die Bezeichnungen 'Universität Duisburg' und 'Universität Essen' nicht mehr. Beide Universitäten führen jetzt den gemeinsamen Namenszug 'Universität Duisburg-Essen' spezifiziert durch den Zusatz 'Campus Essen' bzw. 'Campus Duisburg'.

[2] Im Bereich der Primarstufe deckt das Fach Technik den Lernbereich Sachunterricht Naturwissenschaften-Technik ab.

[3] Mit Ende des WS 02/03 besteht an der Universität Duisburg-Essen, Campus Duisburg, nicht mehr die Möglichkeit einer Immatrikulation für das Lehramt im Fach Technik.

[4] Diese 1981 in Kraft getretenen Richtlinien für das Fach Technik in der gymnasialen Oberstufe enthalten die Regelungen für den Grundkursbereich.

[5] Zu den drei richtlinienkonformen Hauptzielen des Technikunterrichts gehören: technikwissenschaftliche Grundlagen und technikspezifisches Sachwissen; technisches Denken und Handeln; sachgerechtes, reflektiertes und verantwortungsbewusstes Verhalten in durch Technik mitbestimmten Situationen.

- wichtige Disziplinen der Technik (wie z. B. Energietechnik, Verfahrenstechnik, Produktionstechnik, Mess-, Steuerungs- und Regelungstechnik, Nachrichtentechnik, etc.),
- didaktisch/methodische Fragestellungen [Schmayl, 1995; Wilkening, 1996] für den Unterricht und die Unterrichtsverfahren in den einzelnen Schulstufen (Sekundarstufe I und II) und Schulformen (Gymnasium, Gesamtschule, Realschule und Hauptschule).

Eine Betrachtung der beiden für das Fach Technik wesentlichen Schulstufen liefert folgendes Bild:

Sekundarstufe I

Die zunehmende Bedeutung der Technisierung und Industrialisierung erfordert die Fähigkeit zu technischem Denken und Handeln sowie die Kenntnis konkreter technischer Systeme.

- An der Hauptschule ist das Fach Technik Bestandteil des Lernbereichs Arbeitslehre. Dieser umfasst neben dem Fach Technik auch die Fächer Wirtschaftslehre und Hauswirtschaft [Richtlinien, 1989].
- In der Realschule bildet das Fach Technik im differenzierten Unterricht der Jahrgangsstufen 9 und 10 einen Neigungsschwerpunkt, den die Schüler mit dem Übergang in die Klasse 9 wählen können. Dieser Neigungsschwerpunkt besteht aus sechs Themenkreisen[1] [Richtlinien, 1986].
- An der Gesamtschule wird das Fach Technik in den Jahrgangsstufen 5, 7, 9, und 10 unterrichtet [Richtlinien, 1998].
- Das Gymnasium bietet das Fach Technik im Wahlpflichtbereich II der Klassen 9 und 10 mit jeweils drei Unterrichtsstunden an [Richtlinien, 1993].

Sekundarstufe II

Die Entwicklung und Erweiterung technischer Handlungskompetenz ist ein herausragendes Ziel des Technikunterrichts in der Sekundarstufe II. Technische Handlungskompetenz zeigt sich u.a. in der Fähigkeit zur Konzipierung technischer Prozesse zur Umsetzung von Stoffen, Energien und Informationen und darüber hinaus in der Realisierung dieser Prozesse in Apparaten, Maschinen und Geräten, in der Handhabung, Bewertung und letztlich auch in der Entsorgung dieser Systeme[2].

In der gymnasialen Oberstufe wird das Fach Technik in zwei Formen unterrichtet:

- Als *Grundkurs* mit 3 Unterrichtsstunden pro Woche [Richtlinien, 1999].
- Als *Leistungskurs* mit 5 Unterrichtsstunden pro Woche [Richtlinien, 1994].

Als experimentelles Fach gehört es zu der Fächergruppe des mathematisch-naturwissenschaftlich-technischen Aufgabenfeldes (Lernbereich III). Generell ersetzt

[1] Im Einzelnen sind das: Arbeit und Produktion, Transport und Verkehr, Versorgung und Entsorgung, Energie und Maschinen, Information und Kommunikation, Automation.

[2] Technische Handlungskompetenz ist hier auf relevante Inhalte der Sekundarstufe II eingegrenzt.

das Fach Technik nicht die Pflicht zur Teilnahme am naturwissenschaftlichen Unterricht. Es ist z. B. gleichberechtigt mit dem Fach Informatik als zweites oder drittes (schriftlich) bzw. viertes (mündlich) Abiturfach zugelassen [Abiturfach Technik, 2003]. Im Rahmen einer Änderung der Belegungspflicht kann das Fach Technik ab dem Schuljahr 2002/2003 als 2. Pflichtfach im Lernbereich III der gymnasialen Oberstufe in NRW gewählt werden.

Nordrhein-Westfalen war das erste Bundesland, welches das Fach Technik in der Sekundarstufe II einführte. Es folgten 1992 Brandenburg [Potsdam, 2002] und 1999 Sachsen-Anhalt [Sachsen-Anhalt, 2001; Technikdidaktik, 2003]. Seit Anfang des Jahres 2003 wird eine Einführung des Faches Technik im Bundesland Bayern vorbereitet.

Die seit 1988 gültige Studienordnung [StO, 1988] des Faches Technik wurde zum Sommersemester 2000 durch eine neue Studienordnung [StO, 2000], die u.a. eine zeitgemäße Neuordnung des Bereichs "Informationsumsatz in technischen Systemen" aufweist, ersetzt. Fast zur gleichen Zeit entstanden auch überarbeitete Richtlinien für dieses Fach [Richtlinien, 1999], die diesem Umstand ebenfalls Rechnung tragen. Auch die neue Fassung der Lehramtsprüfungsordnung [LPO, 2003] fordert unter §5 (1) den "Erwerb von Fähigkeiten zum fachspezifischen Umgang mit Informations- und Kommunikationstechniken und pädagogische Medienkompetenz", ebenso wie das Rahmenkonzept für medienpädagogische Kompetenz in der Lehrerausbildung, für dessen Umsetzung ein Stufenplan erarbeitet worden ist, der durch "Ausstattung, Fortbildung und personelle Unterstützung sowie durch finanzielle Anreize und durch Projektausschreibungen den systematischen Aufbau von Medienkompetenz in der Lehrerausbildung sicherstellen soll" [Landtag NRW, 2003]. Damit ist das Fach Technik in der Pflicht, seinen Studentinnen und Studenten neben einer allgemeinen informationstechnischen Grundbildung auch eine gezielte fachspezifische Erweiterung im informationstechnologischen Bereich anzubieten. Durch die Einführung der in der neuen Studienordnung verankerten Veranstaltungen Datenverarbeitung III und Datenverarbeitung IV wird dieser Forderung nachgekommen [Wehling, 2003].

Das Lernen im Fach Technik darf nicht zuletzt wegen der Forderungen der Lehramtsprüfungs- und Studienordnung auf den Einsatz von Tafel und Kreide bzw. Overheadprojektor reduziert werden; gerade in seiner Funktion als besonders innovatives Fach hat es sich an den technischen Neuerungen der Zeit zu orientieren:

Das Aufkommen der Neuen Medien[1] verlangt deren angemessenen Einsatz! [TERC, 2002]

Die Studentinnen und Studenten des Faches Technik sollen mit Abschluss ihres Studiums in der Lage sein, als Multiplikatoren und Botschafter des Faches, im Rahmen der ihnen vermittelten informationstechnologischen Kenntnisse, einen adäquaten und kompetenten Einsatz der Neuen Medien im Unterricht zu gewährleisten. Sie werden dabei zusätzlich durch einen eigens für dieses Fach gegründeten Verein (TUF e.V.) unterstützt. Der TUF [TUF, 2003] nimmt als Interessengemeinschaft, in der sich unter anderem Hochschullehrer, Lehrer, Referendare und Studierende zur Förderung des Tech-

[1] Der Begriff Neue Medien bezeichnet hier Medien der Informations- und Kommunikationstechnologie, die mit Hilfe neuer und erweiterter Technologien auf digital- und computertechnischer Basis arbeiten.

nikunterrichts in der gymnasialen Oberstufe zusammengeschlossen haben, eine zentrale Position zur Informationsbeschaffung ein und arbeitet eng mit großen Vereinen und Gesellschaften, wie dem VDI sowie der DGTB e.V. und der EGTB e.V. zusammen [VDI, 2003; DGTB, 2003; EGTB, 1999].

Dies ist um so wichtiger, als für den Technikunterricht an der gymnasialen Oberstufe kein universell einsetzbares Lehrbuch, sondern ausschließlich Spezialliteratur existiert: "Fachkunde Metall" [Fachkunde Metall, 1999], "Thermodynamik für Maschinenbauer" [Geller, 2003], "Einführung in die Elektronik" [Block, 1997] und "Regelungstechnik für Ingenieure" [Reuter, 2002], um nur einige wenige zu nennen. Das Fehlen eines Lehrbuchs ist hier jedoch sinnvoll: Es hat festgeschriebene Inhalte und würde somit ein unzureichendes Maß an Flexibilität bieten, um den sich aktuell ändernden Inhalten dieses Faches Rechnung tragen zu können. Der teilweise immense Einfluss neuerer Technologien (Computer, Internet, Datenbanken, Lichtwellenleiter) würde unberücksichtigt bleiben [Graube, 2002]. Die Online-Nutzung von digitalisiertem Content, die Unterstützung durch Datenbanken, und die Implementierung von e-Learning[1] stellen Teilaspekte dar, die zu einer Lösung dieses Problems führen können. Ein zielgerichteter und angemessener Einsatz der Neuen Medien kann so zu nachhaltigeren Ergebnissen führen, als es durch die alleinige Verwendung von Lehrbüchern möglich wäre [VdS, 2003]. Ein stetig wachsender und relativ unüberschaubarer Markt bietet, bezogen auf die jeweilige Art der Vermittlung sowie deren Umsetzung für ein elektronisch gestütztes Lernen, eine Fülle an Lernsoftware an, deren grundlegende Ausführungen beispielhaft in kurzer Form vorgestellt werden sollen:

- **Computer Based Training[2]** (CBT): Typisches Exponat eines CBT ist die Doppel-CD-ROM "Energiewelten, Wie der Strom in die Steckdose kommt" (Bezugspreis: € 15,29), die energietechnische Zusammenhänge in multimedialer Form zur Verfügung stellt [Energiewelten, 2000]. Sie ist netzwerkfähig, bedingt interaktiv und laut Verlag "eine Informationsquelle zum Thema Strom für jedermann" [HEA, 2003]. Während die erste CD Themen rund um den elektrischen Strom in leicht verständlicher Form mit vielen bunten Graphiken, Animationen und Bildern indexgesteuert anspricht, enthält die zweite CD ein Archiv, in dem alle zugehörigen Graphiken und Fotos abgelegt sind. Eine datenbankgestützte Abfragesystematik ist nicht integriert; auch wird eine fachwissenschaftlich fundierte bzw. zielgruppengerechte Auswahl einzelner Inhalte dem Anwender überlassen.
- **Modular orientierte Software** für das vom Vertreiber nicht näher spezifizierte Fachgebiet "Technik" wird unter der Thematik "Technik begreifen, Wissenschaft zum Anfassen" angeboten [DIM, 2003]. In Zusammenarbeit mit dem Institut für Fachdidaktik Physik und Lehrerbildung (IFPL) der TU Berlin können unter dem Leitsatz "Lernsoftware selbst gestalten" einzelne Module (z. B. Dampfmaschine, Kläranlage, So-

[1] e-Learning kann als Oberbegriff für elektronisch (damit sind vorwiegend internetbasierte Technologien der Informations- und Kommunikationstechnik gemeint) gestütztes Lernen definiert werden. Dabei ist zu berücksichtigen, dass die eingesetzten Technologien unmittelbar mit dem Lernprozess verbunden sind und nicht nur als rudimentäre Hilfsmittel dienen.

[2] Es handelt sich dabei um CD-gebundene Lernsoftware, die den Computer als Medium nutzt. CBT-Systeme stellen eine Ergänzung zu traditionellen Lernmitteln dar und können komplexe Sachverhalte durch Verbindung von Text, Bild, Ton und Videoanimationen veranschaulichen.

larzellen) erworben und durch einen speziellen HTML-Code browserspezifisch in eigene Web-Seiten integriert werden [Physics4You, 2003]. Die Module sind teilweise an gängigen Lehrbüchern der einzelnen Schulstufen orientiert und stellen im wesentlichen Simulationen einzelner Experimente nach (Bezugspreis pro Modul: ab € 0,50). Der Verlag verzichtet auf jegliche zielgruppen- und situationsspezifischen Erläuterungen zu den einzelnen Modulen und überlässt auch in diesem Fall dem Anwender die fachspezifische und zielgruppengerechte Auswahl einzelner Inhalte. Zusätzlich wird ein spezieller Player gebraucht, um mit dieser Software arbeiten zu können.

- **Web Based Training**[1] (WBT): Das von einem spanischen Unternehmen [ITACA, 2003] vertriebene und von der deutschen Presse [TeachersNews, 2003] hoch gelobte "Multimedia Course Authoring Tool EasyProf" ist neben spanischer und englischer auch in deutscher Sprache verfügbar. Es wirbt damit, eigene, unterrichtsrelevante Inhalte ohne nennenswerte Programmierkenntnisse schnell online verfügbar zu machen. In Zusammenarbeit mit der Universitat Polytèchnica de Catalunya, Barcelona, wird so eine Komplettlösung für internetbasierten Unterricht angeboten, die *Experten* helfen soll, Unterrichtseinheiten zu erstellen. EasyProf [EasyProf, 2003] bringt alle Hilfsmittel mit, um Graphiken, Animationen und internetfähige Texte zu erstellen und verfügt über integrierte Möglichkeiten einer standardisierten Evaluation von Unterrichtseinheiten. Mit einem Jahr Support wird diese Software in der Version 2.0 ab € 895,-- vertrieben.
- **Learning Management System** (LMS): Nach einer Erhebung des bn:bwk sind im April 2003 ca. 100 LMS am Markt verfügbar [bm:bwk, 2003]. In jedem LMS sind die Funktionsbereiche *inhaltliche Präsentation*, *Kommunikation*, *Evaluation und Administration* sowie diverse *Werkzeuge* vorhanden. LMS sind die Plattformen, auf denen sich über eine entsprechend ausgelegte Benutzeroberfläche letztlich ein e-Learning abspielt. Ein LMS ist daher nichts anderes, als eine spezielle Software für die Initiierung und Durchführung eines webgestützten Lernens. Das Integrierte Lern-, Informations- und Arbeitskooperations-System der Universität Köln steht exemplarisch für ein intensiv genutztes, kostenloses LMS [ILIAS, 2003].

Diese vier Beispiele beschreiben in kurzer Form die Situation am Markt für Lernsoftware. Die meisten Produkte werden kommerziell vertrieben, wobei die Preisspanne außerordentlich groß ist. Sie bieten fast ausschließlich nicht näher spezifizierten Content an, dem sowohl eine fachwissenschaftliche und fachdidaktische Einordnung, als auch ein zielgruppen- bzw. schul-stufenspezifischer Bezug fehlt. Lernzielorientierte oder lernpsychologische Aspekte bleiben hier ebenfalls vollkommen dem Anwender überlassen.

Es wird an dieser Stelle deutlich, dass viele Multimediaprodukte zwar technisch auf dem neuesten Stand sind, aber sowohl in methodischer als auch in didaktischer Hinsicht eine durchdringende Überarbeitung unumgänglich ist [Groten, 2002]. Bei der Entwicklung multimedialer Lehr- und Lernmittel ist daher eine intensive interdisziplinäre Zusammenarbeit zwischen Technikern, Pädagogen und Organisatoren notwendig [VDI Nachrichten, 2003]. In diesem Zusammenhang muss auch die bisher fehlende spezielle fachliche Ausrichtung solcher Lehr- und Lernmittel verstärkt berücksichtigt werden.

---

[1] WBT ist eine Form des CBT. Der Unterschied liegt darin, dass beim WBT das WAN bzw. ein LAN als Kommunikationsmedium verwendet wird, während CBT-Programme meist als CD-ROM vorliegen.

Selbstverständlich stellen digitalisierte Inhalte in Form einfachen Contents eine notwendige Voraussetzung für das Schaffen komplexer, interaktiver Ressourcen dar. Wenn Content in geeigneter Form zur Verfügung steht, muss er über einen datenbankbasierten Zugang verfügen, der eine systematische Recherche ermöglicht. Deshalb kann eine speziell für das Fach Technik entwickelte *datenbankbasierte Entwicklungsumgebung* [INTEGER, 2003] hier die Voraussetzungen schaffen, um aktuelle und multimediale Inhalte jederzeit abrufbar zu machen und in neu strukturierter Form, angepasst an die jeweiligen Erfordernisse und Schwerpunkte, zur Verfügung zu stellen. Im Einzelnen bedeutet das:

- Einfach aufgebaute **Lernmodule** liefern erste strukturierte Anhaltspunkte für den Einsatz multimedialer Elemente.
- Systematisch strukturierte **Lerneinheiten** liefern Vorschläge für alternative Inhalte und weisen unter Berücksichtigung technikspezifischer Klassifikationen auf Querverbindungen untereinander hin [Richtlinien, 1999a].
- Auf der Basis von Lerneinheiten konzipierte **Lehrgänge** mit lernzielorientierten Kriterien zum Vorlesungs- oder Unterrichtseinsatz [Kroner, 1997] dienen letztlich als Ausgangspunkt für den gezielten Einsatz eines sich anschließenden, bewertbaren e-Learnings in Kommunikations- und Lernplattformen.

Auf diese Art und Weise konzipierte Lernobjekte (Lernmodule, Lerneinheiten und Lehrgänge) liefern die Grundlage, um anschließend durch deren Einsatz in Lern- und Kommunikationsplattformen ein effektives e-Learning bzw. Blended Learning[1] durchzuführen.

Mit der Generierung fachspezifischer Lernobjekte sowie deren fachdidaktischer Ergänzung wird speziell für das Fach Technik durch die *datenbankbasierte Entwicklungsumgebung* ein System nutzbar, das integrativer Bestandteil einer ganzheitlichen Konzeption des Lernens mit multimedialen Inhalten ist [Kapitel 3.2 Forderungen].

2.1 Allgemein

Das Zeitalter der Informationstechnik ist längst angebrochen und birgt eine Fülle von Anreizen und Herausforderungen. Begriffe wie Internet, e-Mail, Webserver, Interaktivität, Multimedia, e-Learning, etc. sind zu einer Selbstverständlichkeit geworden und werden durch das Akronym T.I.M.E. benannt [PATT-11, 2001]. Von allen an diesem Entwicklungsprozess Beteiligten wird eine permanent zu erweiternde mediale Kompetenz erwartet, die den stetig wachsenden Anforderungen unserer informationstechnologischen Umgebungen angepasst erscheint. Der Begriff **Lebenslanges Lernen** stellt eine der stärksten Herausforderungen unserer Zeit, nämlich mit dieser ungeheuer schnellen Entwicklung Schritt zu halten und "mehr in Menschen und ihr Wissen zu investieren, den Erwerb von Basisqualifikationen (einschließlich digitaler Kompetenz) zu unterstützen und den Weg für innovativere, flexiblere Formen des Lernens zu ebnen" [Europa, 2003]. Dies ist gerade deshalb wichtig, weil dieser riesige informationstechnologische

[1] Das Blended Learning kombiniert verschiedene Lernformen und –medien, wobei meistens virtuelle (e-Learning) und nichtvirtuelle (klassische Unterrichtsformen) Methoden zur Unterstützung von Lernprozessen zum Einsatz kommen.

Bereich einer enormen Fluktuation unterliegt: sogenannte dot.com-Firmen entstehen genauso schnell, wie sie verschwinden (e-Commerce) [Heise, 2001]; vielversprechende Innovationen (e-Book) [c't, 2003] sind obsolet, noch bevor sie standardisiert werden können; bahnbrechende Hardware (Satelliten-Telefonnetz) wird vom Enduser nur zögerlich angenommen [Iridium, 2003]; Rechner und Betriebssysteme sind nach drei Jahren eine Antiquität; riesige Firmen etablieren sich am weltweiten Markt als Monopolisten[1] [IBM, 2003; Intel, 2003; Microsoft, 2003].

2.2 Universitär

Förderinitiativen des Bundesministeriums für Bildung und Forschung [HIS, 2003] und (regional) des Ministeriums für Schule, Jugend und Kinder (MSJK) sowie des Ministeriums für Wissenschaft und Forschung (MWF) des Landes Nordrhein-Westfalen [Bildungsportal, 2003] sind konsequente Reaktionen auf die informationstechnologischen Herausforderungen[2]. Länderspezifische Verbundinitiativen zur Förderung der Neuen Medien, regional vertreten durch das Kompetenznetzwerk Universitätsverbund MultiMedia NRW [UVM, 2003], initiieren Projekte und bieten Weiterbildungsseminare, wie z. B. "Urheber- und Nutzungsrecht im Internet" [Multimediarecht, 2001] zur Erweiterung medialer Kompetenzen an. Im Projektumfeld entstehen länderübergreifende [CMMV, 1999] und regionale Clearingstellen [MMCL, 2003], die Projektarbeiten koordinieren und Redundanzen offen legen. Die Rechenzentren der Hochschulen reagieren mit der Anbietung zeitgemäßer Serviceleistungen: innerhalb der Neuen Medien stehen von der digitalen Bibliothek [DigiBib, 2003] über Online-Vorlesungen [Darmstadt, 2003] bis hin zum kompletten Fernstudium [Hagen, 2003] alle Möglichkeiten für den Anwender offen. So ist ein Austausch verfügbarer Projektergebnisse ebenso schnell möglich, wie Studien über Anwendbarkeit, Nutzen, Evaluation, Nachhaltigkeit und Kosten laufender Projekte.

Es ist abzusehen, dass sich der universitäre Vorlesungsbetrieb durch den zunehmenden Einsatz der Neuen Medien stark verändern wird [Schulmeister, 2001]. Hier werden sich vermutlich ökonomische Interessen intensiv in den Vordergrund schieben.

2.3 Fachspezifisch

Das Fach Technik hat neben einer Vermittlung von medialer Kompetenz im besonderen die Aufgabe [Richtlinien, 1999b], zu einem zeitgemäßen, sachgerechten Aufarbeiten und Darbieten von technischen Inhalten, sowie zu einer permanenten Auseinandersetzung mit technischen Innovationen und Neuerungen im Rahmen von Schule und Hochschule anzuregen [MMDB-TU, 2003]. Diese infrastrukturellen Momente implizieren mediale Umsetzungen, die nicht ausschließlich auf den Bereich "Informationsumsatz in technischen Systemen" beschränkt sein müssen.[3]

Seit 1999 wurden und werden im Fach Technik Projekte durchgeführt, die sich alle mit

---

[1] IBM, Intel und Microsoft sind eingetragene Warenzeichen.

[2] Seit dem 12. Nov. 2002 gliedert sich das ehemalige Ministerium für Schule, Wissenschaft und Forschung des Landes Nordrhein-Westfalen (MSWF) in die beiden Ministerien MSJK und MWF.

[3] Hier sei auf die in den Richtlinien dargelegten Ausführungen zu den Aufgaben und Zielen des Faches Technik verwiesen.

dem Problem des Lernens unter Nutzung der Neuen Medien beschäftigen.

- *Projekt 1, Laufzeit vom 01.04.1999 bis 31.03.2001:* Komponentenbasierte Lernsoftware für die Lehrerausbildung (Physik und Technik in den Sekundarstufen I und II sowie Lernbereich Sachunterricht Naturwissenschaften-Technik) und für die integrierten Studiengänge Physik-Diplom I und II [UVM, 2003a]
- *Projekt 2, Laufzeit vom 01.01.2001 bis 31.12.2003:* Entwicklung von Lehr- und Lernmodulen im Baukastenmodus zu den disziplinübergreifenden Bereichen Stoff-, Energie- und Informationsumsetzende Systeme für die Studiengänge Lehrämter Technik Sekundarstufe I und Sekundarstufe II [TeLos, 2003]
- *Projekt 3, Laufzeit vom 01.10.2001 bis 31.12.2003:* Einsatz und Evaluation von Kommunikations- und Lernplattformen zur Verbesserung der Studierbarkeit des Faches Technik[1] [LPF, 2003]

Allen Projekten ist eines gemeinsam: es wird eine Verbesserung der Studierbarkeit des Faches Technik sowie eine Erweiterung der medialen Kompetenz durch gezielten Einsatz der Neuen Medien im Rahmen eines damit verbundenen didaktischen Mehrwertes angestrebt. Genau hier (*Projekt 2*) ist die vorliegende Arbeit angesiedelt.

In diesem Zusammenhang sei darauf hingewiesen, dass das Fach Technik seit Jahren sein Equipment aus dem Bereich der Neuen Medien (z. B. Video-Beamer, Computer, Anwendungssoftware, etc.) regelmäßig ergänzt bzw. erneuert und so zu einer intensiven Einbindung der Neuen Medien in die Bereiche Forschung und Lehre beiträgt.

2.4 Mögliche Szenarien

Ein sehr wahrscheinliches Szenario [Encarnação, 2000] wird durch den Expertenkreis "Hochschulentwicklung durch neue Medien" der Bertelsmann Stiftung [Bertelsmann, 2003] und der Heinz Nixdorf Stiftung [Nixdorf, 2003] dargestellt. Es wird davon ausgegangen, dass spätestens im Jahre 2005 die deutschen Hochschulen in direktem Wettbewerb mit privaten Anbietern stehen werden, wobei mehr als die Hälfte aller Studierenden schon virtuelle Studienangebote nutzen wird. Dann werden Studenten nicht mehr nur zwischen staatlichen Hochschulen wählen können, sondern zusätzlich noch private Bildungsanbieter und sog. Corporate Universities [Kraemer, 2001] einbeziehen können. Hier werden Kooperationen zwischen Hochschulen und Wirtschaftsunternehmen sowie Bildungs-Broker für individuelle Bildungsangebote neue Akzente in der Bildungslandschaft setzen [Zentel, 2002].

Innovativer und/oder effizienter Multimedia-Einsatz wird eine immer stärkere Vorbildfunktion innerhalb bildungsspezifischer Projekte einnehmen. Diese Tendenz ist bereits deutlich zu beobachten, wobei außerdem spezielle Entwicklungstrends durch Multimedia (Lernen via interaktivem TV, CD-ROM, Videobänder, etc.) und bereits erprobten Entwicklungen (Kombination von Präsenzlehre mit Informations- und Kommunikationstechnik in multimedialen Lehr- und Lernsystemen, Tele-Lecturing) [MZ, 2003] berücksichtigt werden müssen [EQ, 2002].

---

[1] Zu diesem Zweck stehen über Projektmittel 12 Laptops zur Verfügung, die über ein funkgestütztes Netzwerk verbunden sind.

Wenn die Hochschulen im Zeitalter eines globalen Online-Bildungsmarktes konkurrenzfähig bleiben wollen, müssen sie sich den Anforderungen des Informationszeitalters stellen. Die Hochschulen in Deutschland werden dabei im direkten Wettbewerb mit internationalen und privaten Anbietern stehen. Dabei sind effektive Entwicklungsstrategien mit entsprechendem Management auf Organisationsebene sowie mediengestützte Lernszenarien auf Hochschulebene unerlässlich [GMW, 2003].

Allerdings sollte unmissverständlich klar sein, dass private Bildungsanbieter bezahlt werden müssen, und zwar letztlich von Seiten der Studenten. So muss beispielsweise schon heute der Studierende für einen Master of Business Informatics an der Virtual Global University pro Semester rund € 5000 bezahlen [Meyer, 2003].

3 Problemstellung

Hier soll der Ansatz, der zu einer Verbesserung der Studierfähigkeit durch Erweiterung der medialen Kompetenzen hinleiten kann, problematisiert werden. Dabei steht der adäquate Umgang mit den Neuen Medien im Mittelpunkt, da er untrennbar mit Begriffen der informationstechnischen Grundbildung, wie z. B. Umgang mit Computern, Betriebssystemen und An-wendungsprogrammen sowie Kenntnissen über Einsatzmöglichkeiten und Kontrolle von Informationstechniken verbunden ist.

Neben Förderprojekten des Bundes und der Länder [BMBF, 2003] wird eine europaweite Projektarbeit [DAAD, 2003] immer stärker favorisiert, um z. B. synergetische Effekte zu erzeugen bzw. zu nutzen. In diesem Umfeld wandeln sich die Kernaussagen von CBT über WBT hin zu e-Learning und Blended Learning, wobei der Aspekt der netzgebundenen Interaktivität immer stärker akzentuiert wird [Niegemann, 2001].

Bei einer Intensivierung der informationstechnischen Grundbildung mit dem Schwerpunkt Neue Medien sind didaktische, methodische und somit auch lernzielrelevante und lernpsychologische Aspekte ausdrücklich zu berücksichtigen [Aufenanger, 2002]. Ebenfalls müssen die Besonderheiten und Rahmenbedingungen dessen, was für ein Lernen mit multimedialen Lernobjekten relevant ist, eingeschlossen sein, d. h. jedes Fachgebiet besitzt sein spezielles Charakteristikum, das exakt beachtet werden muss. So hat z. B. das Experiment in der Physik einen gänzlich anderen Stellenwert als in der Technik: während im naturwissenschaftlichen Bereich Naturkonstanten nachgewiesen werden (z. B. Messung der Elektronenladung), liegt der Schwerpunkt im ingenieurwissenschaftlichen Bereich im experimentellen Test eines Systems (z. B. Prüfung der Dauerschwingfestigkeit, Optimierungsexperiment durch Parametervariation). Dieser Unterschied ist beispielsweise bei einer Entwicklung von Simulationen besonders zu berücksichtigen.

Lern- und Kommunikationsplattformen wie WebCT [WebCT, 2003], Blackboard [Blackboard, 2003] oder OpenUSS [OpenUSS, 2003], um nur einige wenige zu nennen, weisen den richtigen Weg, sind aber nicht universell und daher auch nicht ohne Einschränkung in jedem Fachgebiet einsetzbar: so sind z. B. WebCT und Blackboard[1] teure, kommerziell vertriebene Produkte, während OpenUSS als Freeware ausschließlich in deutscher Sprache vorliegt und daher für einen länderübergreifenden Einsatz ungeeignet ist.

Mit Bezug auf Kapitel 2 sind die verfügbaren Medien für ein Lernen mit elektronischen Mitteln stark durch die Art der jeweiligen Vermittlung geprägt (CBT, WBT, LMS, etc.). So ist beispielsweise keine der auf dem Markt erhältlichen Lern- und Kommunikationsplattformen in der Lage, **alle** speziellen Ansprüche[2] zu erfüllen [Baumgartner, 2002].

---

[1] Die Nutzung von WebCT und Blackboard ist lizensiert. Die Lizenzgebühren (in US-Dollar) sind abhängig von der Zahl der Studierenden und dem Zeitraum der Anwendung. WebCT: Standard Edition, ca. $ 7.000; Campus Edition, ca. $ 15.000 für bis zu 2.500 Nutzer. Bei Blackboard liegen die Lizenzgebühren in einer ähnlichen Größenordnung.

[2] Blackboard bietet z. B. keine Möglichkeit der Evaluation, d. h. es fehlt die Möglichkeit einer Überprüfung des Lernerfolgs!

Existierender Content sollte in einem Pool fachspezifisch strukturiert jedem Nutzer zur Verfügung stehen und neben den Möglichkeiten einer Recherche auch die Möglichkeiten einer themenspezifischen Umstrukturierung eröffnen. Auf dieser Grundlage ließen sich schließlich Inhalte vorbereiten, die den Einsatz einer Lern- und Kommunikationsplattform sinnvoll erscheinen lassen [Bhattacharya, 2001].

Eine themenspezifische Umstrukturierung von Content lässt sich durch einen modularen Ansatz realisieren [Kapitel 4.1, Modularer Ansatz], wobei zusätzlich ein gestuftes Lernen mit multimedialen Inhalten [Kapitel 3.2 Forderungen] dem eines Lernens durch den ausschließlichen Einsatz von Lern- und Kommunikationsplattformen vorzuziehen ist.

Daraus ergibt sich die folgende Kernfrage:

Wie lässt sich auf der Basis existierenden, unstrukturierten[1] Contents ein nachhaltiges, fachwissenschaftlich einwandfreies und didaktisch begründetes, multimediales Lernen unter angemessenem Einsatz der Neuen Medien unterstützen?

Um diese Kernfrage zu beantworten und daraus begründete Forderungen abzuleiten, ist jedoch zuerst eine Klärung der damit verbundenen Begrifflichkeiten im Rahmen der Neuen Medien notwendig.

3.1 Multimediales Lernen

Der Begriff des multimedialen Lernens unter Einbeziehung der Neuen Medien erfordert eine begriffliche Definition von Multimedia als integrativen Bestandteil der Neuen Medien.

Multimedia kann im wesentlichen durch vier Merkmale charakterisiert werden [Blumstengel, 1998]:

- *Digitalisierung von Content:* Die Speicherung und Bearbeitung von auf unterschiedlichen analogen Trägern vorliegenden Daten (Ton-, Bild-, Filmmaterial, etc.) erfolgt auf der Basis digitalisierender Techniken. Hierbei ist zu berücksichtigen, dass eine Digitalisierung immer mit einer Codierung verbunden ist. Die Digitalisierung besteht dabei aus den zwei Schritten der Diskretisierung[2] und der Quantisierung[3] [Hesselmann, 1987].
- *Computerbasierte Integration von Content:* Der Computer ist als Instrumentarium der Digitalisierung von in analoger Form vorliegenden Daten integrativer Bestandteil des Arbeitens mit unterschiedlichen Medientypen[4] [FILExt, 2003], während die contentorientierte Strukturierung durch den Einsatz von Datenbanken erfolgt. Hierdurch wird eine Aufbewahrung an räumlich getrennten Orten möglich, wobei die hochgra-

[1] Unstrukturiert bezeichnet hier das Fehlen einer fachspezifischen Struktur.

[2] Die Diskretisierung ist ein Vorgang der Aufzeichnung von räumlich und/oder zeitlich äquidistanten Messwerten eines analogen Signals.

[3] Als Quantisierung wird die Darstellung einer Messwerte-Abtastung mit endlicher Auflösung, abgebildet auf ganzzahlige Binärwerte bezeichnet.

[4] Unterschiedliche Medientypen sind durch die ihnen eigene Dateiendung gekennzeichnet: Beispiele sind: unformatierte Texte (*.txt), Fotos mit 16,7 Mio. Farben (*.jpg), Hypertext-Dokumente (*.htm, *.html), etc.

dige Vernetzung von Informationsquellen eine einfache Beförderung abstrakter Informationen erlaubt [Henning, 2001].

- *Multimodale und multicodale Präsentation von Content:* Neben dem Ansprechen mehrerer wahrnehmender Sinnesorgane (multimodal) wie Sehen und Hören wird darunter die Verwendung unterschiedlicher Zeichen- und Symbolsysteme, also die Verwendung verschiedener Formate zur Codierung und Encodierung (multicodal) von Information verstanden [Kerres, 2001].
- *Anwendergesteuerte interaktive Nutzung von Content:* Im Zusammenhang mit der ständigen Darstellung relevanter Objekte bieten sich Möglichkeiten einer direkten physischen Manipulation der Objekte sowie der Ausführung inkrementeller und reversibler Operationen, wobei das Ergebnis direkt sichtbar wird [Haack, 1997].

Ein multimediales Lernen unter Einsatz der Neuen Medien ist eng mit Lern- und Kommunikationsplattformen zum e-Learning oder Blended Learning verbunden. Eine Lern- und Kommunikationsplattform, ganz gleich, ob es sich um ein VLE, ein IDLE oder ein LMS handelt, muss über festgeschriebene Mindestkriterien verfügen [bildung.at, 2003]. Das sind u.a.:

- Integrierte Kommunikationstools, wie Diskussionsforen oder Messaging-Systeme
- Trennung von Inhalt und Form
- Unterstützung von Unicode[1] [Unicode, 2003]
- Integrierte Suchfunktionalität
- Gruppenbasierte Benutzerverwaltung und Rechtevergabe
- Integrierter Online-Editor mit WYSIWYG-Funktionalität[2]
- Getrennte Verwaltung einzelner Assets[3]
- Up- und Download-Fähigkeit von Dateien
- Integriertes browserbasiertes Frontend[4] ohne proprietäre Erweiterungen
- Einbindung gängiger MIME-Typen[5]
- Modularer Aufbau und Möglichkeiten einer modularen Erweiterbarkeit
- Mehrsprachigkeit.

Neben diesen, für jede Lern- und Kommunikationsplattform festgeschriebenen Merkmalen, kann ein Lernen nur unter Berücksichtigung allgemein anerkannter lernzielorientierter [Heimann, 1979] bzw. kompetenzorientierter [Bonsen, 2002] und lernpsychologisch [Joerger, 1989; Spitzer, 2000] begründeter Kriterien erfolgen. Selbst unter Berücksichti-

---

[1] Durch das Unicode-System werden Zeichen oder Elemente aller bekannten Schriftkulturen/Zeichensysteme standardisiert.

[2] Durch eine WYSIWYG-Funktionalität wird eine graphische Visualisierung des zu erwartenden Ergebnisses umgesetzt.

[3] Assets sind typische Inhalte von Webseiten, wie z. B. Bilder, Texte, Graphiken, Audio- oder Videodateien, etc.; Assets werden in dieser Arbeit als "Objekte" bezeichnet.

[4] Ein Frontend ist eine automatisch angepasste, graphische Benutzeroberfläche.

[5] MIME-Typen sind ein Internet-Standard für verschiedene Dateitypen (*.jpg, *.htm, *.swf, etc.).

gung der jüngsten Diskussionen, die eine "Lernzielorientierung"[1] für nicht mehr zeitgemäß halten und dem sog. "Selbstregulierten Lernen" auf Basis eines konstruktivistischen Ansatzes [Siebert, 2003] eine bevorzugte Stellung zuschreiben, gibt es dennoch einige bewährte, grundsätzliche Gegebenheiten, die selbst bei einem Lernen unter Berücksichtigung der Neuen Medien ausgesprochen hilfreich sein können[2]. Es würde zu weit führen, didaktische Schwerpunkte der Analyse und Planung auf der Grundlage der unterschiedlichen Didaktischen Modelle vorzustellen [Gudjons, 2002; VIB, 1999].

Die deutliche Lernzielorientierung der Richtlinien[3] verpflichtet auch bei der Planung eines multimedialen Lernens zur Berücksichtigung entsprechender pädagogischer Intentionen. Daher soll an dieser Stelle vorwiegend ein Bezug zur "Lernzielorientierten Didaktik", deren grundlegende Bestandteile von Wolfgang Klafki in die Kritisch-konstruktive Didaktik [Klafki, 1995] mit aufgenommen wurden, hergestellt werden. Gerade im Hinblick auf die Generierung von Lernobjekten (Lernmodule, Lerneinheiten, Lehrgänge) bedarf es einer eindeutigen Formulierung dessen, **was** gelernt werden soll. Dafür sollen die folgenden Begrifflichkeiten eine angemessene und überschaubare Basis darstellen:

- *Lernzielorientierung:* Dieser Begriff steht hier stellvertretend für die von Klafki formulierte "Zielorientierte Inhaltsentscheidung". Klafki spricht in diesem Zusammenhang vom Primat der Zielentscheidung [Dohnke, 2002]. Hierbei wird die Festlegung eines Themas oder einer Thematik aus der bloßen Formulierung des Gegenstandes weggerückt und mit einem für den Gegenstand relevanten Zielaspekt verbunden. Konkret bedeutet das: erst, wenn ein Gegenstand unter einer als pädagogisch relevant erachteten Fragestellung für eine Behandlung in einer Veranstaltung ausgewählt wird, wird er zum Thema. Im Begriff des Themas ist somit neben dem Gegenstand auch der Zielaspekt, unter welchem der gewählte Gegen-stand behandelt wird, enthalten. Einfach gesagt: bei der Formulierung eines Themas ist neben dem **was** (Gegenstand) auch ein **warum** (Zielaspekt) notwendig!
- *Lernzieltaxonomie:* Ein lernzielorientierter Ansatz ist immer auf eine Verhaltensänderung der Zielpersonen in den Bereichen der Kognition, des Affekts und der Psychomotorik ausgerichtet [Mager, 1994]. In den Richtlinien des Faches Technik wird dieser implikative Zusammenhang besonders verdeutlicht durch Kompetenzbeschreibungen, wie z. B. "...Fähigkeit und Bereitschaft, Problemstellungen selbstständig...zu bearbeiten...", "...Fähigkeit und Bereitschaft, Entwicklungschancen und Einschränkungen...zu reflektieren und zu beurteilen...", "die Fähigkeit und Bereitschaft, soziale Bindungen und Konflikte zu erfassen und zu verstehen..." [Richtlinien, 1999c]. Durch den Begriff der **Fähigkeit** wird die kognitive und psychomotorische Dimension angesprochen; der Begriff der **Bereitschaft** verweist auf die affektive Dimension. Hier wird deutlich, dass sich Lernziele aus diesen drei Dimensionen nicht ausschlie-

[1] Der Begriff der Lernzielorientierung ist eng mit dem Modell einer "Curricularen Didaktik" verknüpft. Der Begriff des Lernziels bleibt hierbei jedoch unangetastet.

[2] An dieser Stelle sei nochmals darauf hingewiesen, dass diese Arbeit sich nicht mit der Erstellung und Bewertung von Lern- und Kommunikationsplattformen, sondern mit dem strukturierten Ablegen und neu strukturierten Kombinieren von Lernobjekten beschäftigt.

[3] Hier sind die Richtlinien im allgemeinen, die des Faches Technik jedoch im besonderen gemeint.

ßen, sondern gegenseitig bedingen. Die hier genannten drei Dimensionen sind hierarchisch geordnet [Bloom, 1984]:

Kognitive Dimension: Sie bezieht sich auf Denken, Wissen, Problemlösen, auf Kenntnisse und intellektuelle Fähigkeiten, wobei die Lernziele dieser Dimension nach dem Grad ihrer **Komplexität** aufsteigend hierarchisch geordnet sind: Kenntnis, Verständnis, Anwendung, Analyse, Synthese, Beurteilung.

Affektive Dimension: Hier geht es um die Veränderung von Interessenslagen, um die Bereitschaft, etwas zu tun oder zu reflektieren, um Einstellungen und Werte, sowie um die Entwicklung dauerhafter Werthaltungen. Lernziele dieser Dimension sind nach dem Grad der zu bewirkenden **Internalisation** ebenfalls aufsteigend hierarchisch geordnet: Beachten, Reagieren, Werten, Organisieren, Charakterisierung durch eine Wertstruktur.

Psychomotorische Dimension: Hiermit sind die manipulativen und psychomotorischen Fertigkeiten gemeint. Sie sind nach dem Grad der **Koordination** wiederum aufsteigend hierarchisch geordnet: Imitation, Manipulation, Präzision, Handlungsgliederung, Naturalisierung.

Insbesondere Arbeiten zur kognitiven Dimension wurden von Benjamin S. Bloom schon in den 50-er Jahren publiziert. Lernzielformulierungen in der affektiven Dimension gehen im wesentlichen auf David R. Krathwohl, die in der psychomotorischen Dimension auf J. P. Guilford zurück [Krathwohl, 1978; Guilford, 1971]. Die damals entwickelte Forderung nach einer Dimensionierung und Hierarchisierung von Lernzielen ist bis heute anerkannt. Selbstverständlich ließen sich noch wesentlich mehr Kriterien anführen, die für ein effektives Lernen eine hohe Relevanz hätten, jedoch ist das eine Aufgabe, der sich die Entwickler von Kommunikations- und Lernplattformen zur Verwirklichung eines e-Learnings oder Blended Learnings auch im Rahmen eines konstruktivistischen Ansatzes zunehmend werden widmen müssen.

- *Schwerpunktlernziel:* Durch die Angabe eines Schwerpunktlernziels erfolgt eine Konkretisierung des in der Themenformulierung enthaltenen Zielaspekts. Hiermit soll verdeutlicht werden, was über die Beschäftigung mit dem Gegenstand hinaus gelernt und als **transferierbar** bereitgestellt werden soll. Dieser Lernzuwachs kann durch die Formulierung weiterer untergeordneter Lernziele (Feinziele) legitimiert werden.

In jedem Fall ist bei einer Formulierung von Lernzielen darauf zu achten, worin der didaktische Mehrwert (Zeitökonomie, Motivation, Kommunikation, Evaluation, etc.) beim e-Learning besteht, oder ob die angestrebten Lernziele nicht auch mit anderen, insbesondere traditionellen Methoden erreicht werden können.

3.1.1 Content

Digitalisierter Inhalt (kurz: Content) kann in den unterschiedlichsten Formaten vorliegen, wobei sich die Formate über den zugehörigen MIME-Typ bzw. über ihre Datei-Endungen identifizieren lassen. Schon bei alleiniger Verwendung von lediglich 3 Kleinbuchstaben für eine Dateiendung ergeben sich 26^3 verschiedene Kombinationen, also weit über 17.000 verschiedene Dateiformate. Diese Vielschichtigkeit bedeutet, dass temporäre, multimediale, webbasierte, proprietäre, antiquierte und nicht standardisierte Formate enthalten sind.

Noch bis vor kurzem war es gängige Praxis, im Rahmen der Nutzung der Neuen Medien unstrukturiert Content zu produzieren und dem Anwender, ganz gleich auf welche Weise, zur Verfügung zu stellen. Erst allmählich setzt sich die Erkenntnis durch, dass neben speziellen fachwissenschaftlichen Forderungen (technisch-produktorientiert) zunehmend fachdidaktische Kriterien (pädagogisch-prozessorientiert) an Bedeutung gewinnen [Baumgartner, 2002a].

Dieser Umstand führt auf direktem Weg in die Diskussion einer Standardisierung bei der Entwicklung von Content. Hierbei ist allerdings zu berücksichtigen, dass jeder zu entwickelnde Content eine sehr spezielle Zielgruppe hat, wobei zusätzlich anwendungsbezogene und lernpsychologische Aspekte mit einbezogen werden müssen. Es reicht z. B. nicht aus, ein kommerzielles Anwendungsprogramm zur Erzeugung von Simulationen (z. B. Flash) auf der Programmierebene zu beherrschen; es muss dem Programmierer vielmehr klar sein,

- in welcher Fachdisziplin,
- bei welcher Klientel und
- unter welchem Lernziel

ein zu entwickelnder Content eingesetzt werden soll. Erst dann wird der Programmierer in der Lage sein, sinnvolle interaktive Elemente in einem multimedialen Rahmen zu programmieren. Diese Entscheidungen können jedoch nur (s.o.) durch einen Professionalisierungsprozess im Rahmen einer sinnvollen Zusammenarbeit von Programmierern, Fachwissenschaftlern und Fachdidaktikern bewältigt werden [Bentlage, 2002].

In diesem Zusammenhang sind neben den anfallenden Kosten für die Erstellung von Content noch Fragen nach dem Urheber- und Nutzungsrecht [Multimediarecht, 2001] zu stellen, die im Rahmen der Neuen Medien noch längst nicht geklärt sind. So vergeben beispielsweise alle Verlage für Kopien aus Büchern, die im Schulunterricht eingesetzt werden, automatisch die Nutzungsrechte (§46 UrhG-E); anders ist die Situation bei den Universitäten (§52a UrhG-E) [Fraunhofer, 2002]. Hier muss ein Nutzungsrecht in jedem einzelnen Fall beim entsprechenden Verlag beantragt werden [Junker, 1999]. Eine Novellierung des Gesetzes zur "Regelung des Urheberrechts in der Informationsgesellschaft" verzögert sich aufgrund von Meinungsverschiedenheiten zwischen Bundestag und Bundesrat, wobei es vorwiegend um den Urheberschutz privater Normenwerke und um das Recht zur Privatkopie geht [IUM, 2003].

Da für viele sehr gut in Vorlesungen oder Seminaren einsetzbare Inhalte die Nutzungsrechte nicht nur über den Verlag, sondern auch von daran angeschlossenen Dienstleistungsunternehmen bis hin zum Urheber verteilt sein können, ergibt sich eine Situation, auf die kurzfristig nur in einer Form reagiert werden kann:

Jeglicher Content muss in Eigenarbeit produziert werden!

Nur so kann gewährleistet werden, dass ein Urheber die Nutzungsrechte innerhalb einer geschlossenen Gemeinschaft (Community) nach Belieben verteilen kann. Eine solche Community kann beispielsweise der Zusammenschluss aller registrierten Nutzer der *datenbankbasierten Entwicklungsumgebung* sein.

3.1.2 Datenbanken

Rein begrifflich betrachtet lässt sich eine 'Datenbank' unterschiedlich definieren: eine Datenbank kann (a) ein Teil einer Software sein, um Daten strukturiert zu verwalten, es kann sich (b) um ein text- oder graphikbasiertes Frontend handeln, mit welchem man auf strukturierte Datensätze zugreifen kann und schließlich kann der Begriff auch noch für (c) den digitalisierten Datenbestand an sich stehen. Durch die Integration von Frontend und Backend[1] verwischen etablierte Software-Hersteller wie Microsoft[2] oder Corel[3] diese begrifflichen Definitionen [Schulz, 2003].

Relationale DBMS, kurz RDBMS[4] genannt, sind innerhalb der Nutzergemeinschaft wesentlich verbreiteter, als objektorientierte DBMS[5] und werden daher besser durch Newsgroups[6] oder Hersteller-Support unterstützt. Neben kommerziellen RDBMS wie z. B. DB2, Informix, Oracle, Paradox, Filemaker, etc. gibt es leider nur wenige RDBMS, die im Rahmen der GNU GPL frei verfügbar sind: z. B. MySQL [Dubois, 2000], PostgreSQL [Hartwig, 2001], Interbase[7] [Brand, 2002] und SAP DB[8] [Banke, 2002], wobei gerade die freie Verfügbarkeit eine nähere Betrachtung notwendig macht.

Grundlage für alle datenbankgebundenen Operationen ist die Programmiersprache SQL [Matthiessen, 2000]. SQL-orientierte DBMS speichern Daten nach festgelegten Kriterien. Es sind aktive Datenbanken, die in der Regel über Triggerfunktionen[9] verfügen und sicherstellen, dass die Integrität der gespeicherten Daten nicht verletzt wird. SQL ist als Programmiersprache international standardisiert. Ihr Befehlssatz wird in unregelmäßigen Abständen aktualisiert und erweitert. Viele SQL-orientierte DBMS richten sich nach dem 1992 geschaffenen Standard ANSI-SQL/92 [Date, 1998], jedoch nutzen immer mehr DBMS proprietäre SQL-Dialekte bei der Behandlung der von ihnen zu verwaltenden Datensätze. Es bleibt anzumerken, dass dieser Standard 1999 überarbeitet wurde (ANSI-SQL/99), jedoch immer noch nicht einheitlich genutzt wird. Der zunehmende Einsatz proprietärer SQL-Dialekte ist für kleinere Entitäten[10] mit geringem Datenaufkommen relativ unerheblich, entwickelt sich bei komplexen Anwendungen mit starker Nutzer-Frequentierung jedoch zu einem ernsthaften Problem.

---

[1] Backend und Frontend bilden zwei logisch voneinander zu trennende Schichten: die Benutzeroberfläche und die eigentliche Datenhaltung, womit letztlich der Datenbankserver gemeint ist.

[2] Microsoft ist das eingetragene Warenzeichen der Microsoft Corporation.

[3] Corel ist das eingetragene Warenzeichen der Corel Corporation.

[4] Relationale Datenbankserver verwalten Datensätze durch miteinander verknüpfte Tabellen (Relationen).

[5] Die Spezifikation einer offenen Architektur für verteilte objektorientierte Anwendungen ist CORBA; durch den Einsatz von EJB-orientierten Elementen tritt CORBA jedoch zunehmend in den Hintergrund.

[6] Z. B.: **de.comp.datenbanken.mysql** erreichbar über die Perl-Community http://www.perlunity.de/perl/community/newsgr.shtml .

[7] Interbase wird hauptsächlich von Entwicklern eingesetzt, die mit Werkzeugen der Firma Borland arbeiten, und wird aus diesem Grund nicht näher betrachtet.

[8] Der Betrieb der SAP DB unter einem UNIX-Derivat setzt zusätzlich die Administration durch einen Windows-Rechner voraus. Daher wird dieser Datenbankserver ebenfalls nicht näher betrachtet. Letzte Entwicklungen deuten jedoch auf einen Zusammenschluss des Datenbank-Entwicklers MySQL und SAP hin, um eine gemeinsame Unternehmensdatenbank zu entwickeln [SAPDB, 2003].

[9] Trigger erlauben die Aktivierung einer gespeicherten Prozedur bei Eintreten eines bestimmten Ereignisses.

[10] Eine Entität ist ein Objekt, über welches Informationen in einer Datenbank angelegt werden.

Oracle nutzt eine eigens entwickelte Programmiersprache (PL/SQL), Informix arbeitet mit SPL, Microsoft und Sybase unterstützen T-SQL, während DB2 von IBM einen ganz anderen Ansatz verfolgt und Programmiersprachen wie Rexx, COBOL, C und Java favorisiert. PL/SQL, T-SQL, und SPL sind sog. "block-structured languages" mit eigenen Regeln und einer eigenen Syntax; sie sind aus diesem Grunde bzgl. unterschiedlicher DBMS nicht portierbar [North, 1999].

Seit kurzer Zeit erfährt SQL eine breite Unterstützung innerhalb der Entwickler-Gemeinschaft von Java-orientierten DBMS, wobei die Portierbarkeit durch die Verwendung von Java als einer vom Betriebssystem unabhängigen Programmiersprache für alle auf dem Gebiet der DBMS führenden Unternehmen wichtig ist, da alle spezielle SQL-Server für die verschiedenen Betriebssysteme anbieten. Hier werden jedoch vorwiegend kommerzielle Datenbanken angesprochen: Oracle, IBM DB2, Sybase und Informix [Enosh, 1999].

DB2 beispielsweise stellt ein digitales Klassifikationssystem dar und ist deshalb hervorragend für eine Verwendung in digitalen Bibliotheken geeignet, weniger jedoch als multimedialer Lehr- und Lernserver [MILESS, 2002]. Es beschränkt sich auf die Elementarfunktionen *add* und *access*, wobei ihm jegliche Interaktivität (*execute/process*) fehlt: so stellt es z. B. keine Schnittstelle für PHP zur Verfügung, während aber ein explizit ausgeführter Befehlssatz für ein Ansprechen der DBI-Schnittstelle, über die sowohl MySQL als auch PostgreSQL verfügen, unter PHP vorhanden ist [Krause, 2000]; gerade für die Skriptsprache Perl werden über das Comprehensive Perl Archive Network [CPAN, 2003] Module zur Programmierung der DBI-Schnittstelle angeboten, denn DBI ist die standardisierte Datenbankschnittstelle unter Perl. Es ist daher lohnend, einen kurzen Vergleich zwischen den beiden Open-Source-Produkten PostgreSQL und MySQL vorzunehmen, da sich beide als mögliche Kandidaten für ein RDBMS empfehlen.

Während viele Funktionen von MySQL nicht Bestandteil der ANSI-SQL/92-Standardisierung sind, hält sich PostgreSQL weitgehend an diesen Standard. Leider existieren für PostgreSQL nur ausgesprochen wenige brauchbare Dokumentationen [Geschwinde, 2002]. Hier kann MySQL einen entscheidenden Vorteil durch seine weite Verbreitung verbuchen [Kofler, 2001; Throll, 2002; Maslakowski, 2000]. Für Internetanwendungen und bei Linux-Anwendern[1] ist dieses RDBMS die erste Wahl, denn in diesem Fall ist publiziertes Wissen in den unterschiedlichsten Formen reichlich vorhanden (Internet, Foren, Usenet, Newsgroups, etc.). Leider kennt MySQL u.a. weder Stored Procedures[2] noch Trigger, weder Foreign Keys[3] noch Transactions[4] oder Subselects[5], was dieses RDBMS in Richtung komplexer Anwendungen einschränkt [Kirsch, 2003]. Für einfache Anwendungen mit guter Performance bleibt MySQL, nicht zuletzt wegen der vielen extern verfügbaren Werkzeuge und Hilfsmittel [phpMyAdmin, 2003; MySQL-Front,

[1] Linux das eingetragene Warenzeichen von Linus Torvalds.

[2] Stored Procedures ist compilierter und im Server gespeicherter SQL-Code.

[3] Mit Foreign Keys wird festgelegt, dass der Schlüssel einer Entitätsmenge (Oberbegriff für gleichartige beschreibbare individuelle Elemente) in einer bestimmten Beziehung zum Schlüssel einer anderen Entitätsmenge steht.

[4] Transactions sind in sich geschlossene Folgen von SQL-Anweisungen, die ohne irgendeine Beeinflussung anderer Clients ausgeführt werden können.

[5] Subselects sind Select-Anweisungen, die wiederum in Select-Anweisungen eingebettet sind.

2003] jedoch das einschlägige RDBMS für Linux-Systeme. Für Programme mit komplizierten Zugriffswegen und Datenstrukturen sowie extrem vielen Datensätzen jedoch kommt im Open-Source-Bereich nur PostgreSQL in Frage [Mischke, 2002].

3.1.3 Strukturierte Datensätze

Eine sinnvolle Beschreibung von Objekten durch strukturierte Datensätze, um relevante Informationen auch nach langer Zeit sicher wieder zu finden, ist ein uraltes und offensichtlich immer noch nicht befriedigend gelöstes Problem. Ein Beispiel: "Mitte der 90er Jahre waren mehr als 1,2 Millionen Magnetbänder mit Daten aus 30 Jahren Raumfahrt nicht mehr nutzbar, teilweise wegen mangelnder Zuordnung zu den bisherigen Weltraummissionen und Projekten. Vom sogenannten 'NASA-Effekt' wird gesprochen – die Bänder waren nicht oder nur notdürftig beschriftet" [Rathje, 2002]. Abgesehen von nicht ausgereiften Beschreibungsmöglichkeiten für Datensätze bedürfen die Problematiken der Interoperabilität[1] [Day, 2002] und Portabilität[2] [Micsik, 1999] ebenfalls einer langfristigen Lösung.

Objekt-Beschreibungen durch Metadaten bieten eine mögliche Basis zur Lösung dieser Probleme an. Für Metadaten, die den Kriterien Interoperabilität und Portabilität genügen sollen, sind jedoch sowohl Syntax[3] als auch Semantik[4] wichtig.

Leider existieren unterschiedlichste Konzepte für eine Implementierung von Metadatensätzen für die verschiedensten Bereiche, wobei hier lediglich einige der wichtigsten kurz angesprochen werden sollen [Holzinger, 2001; Weichselbraun, 2002]:

- (X)HTML-Metatags[5]
- Dublin Core Metadata Initiative[6]
- Resource Description Framework[7]
- IEEE Learning Objects Metadata
- Multimedia Content Description Interface.

Keines der hier genannten Metadaten-Konzepte ist universell geeignet; alle sind auf bestimmte Einsatzgebiete zugeschnitten:

(X)HTML-Metatags liefern eine Metadaten-Struktur, die zwar von den meisten internetbasierten Suchmaschinen indexiert wird, aber für eine tiefgehende Beschreibung von

---

[1] Interoperabilität ist die Fähigkeit, Informationen über gemeinsam nutzbare Datenformate auszutauschen.

[2] Portabilität ist die Lauffähigkeit von Anwendungen auf unterschiedlichen Systemen, womit sowohl verschiedene Betriebssysteme als auch diverse Hardware-Architekturen gemeint sind.

[3] Die Syntax gibt an, in welcher Form Metadaten ausgetauscht werden.

[4] Die Semantik beschreibt, welche Metadaten in einem Beschreibungssatz eingesetzt werden können, und welche nicht.

[5] XHTML bietet auf der Grundlage von HTML 4.0 in Richtung XML weisende erweiterte Spezifikationen an.

[6] Weitere Konzepte, wie z. B. GEM, Warwick-Framework und IMS verwenden alle in unterschiedlicher Komplexität, Erweiterbarkeit und Tiefe einfache Beschreibungselemente des Dublin Core.

[7] Das Konzept PICS war ein Vorläufer des RDF und stellte einen einfachen Metadaten-Mechanismus dar, um Webinhalte zu bewerten.

Lernobjekten unzureichend ist. Ein Ausweg bietet sich hier über eine Verwendung von XML[1] an. Leider ist XML kompliziert und kann keine Webseiten darstellen, sondern lediglich deren gewünschte Grundstruktur beschreiben, wobei die endgültige Umsetzung durch HTML-Derivate, StyleSheets, Skriptsprachen und externe Dateien erfolgen muss.

Die *Dublin Core Metadata Initiative* (DCMI) [DCMI, 2002] liefert einen Set von 15 Basiselementen zur strukturierten Beschreibung von Objekten [Hillmann, 2001]. Die Einbindung in HTML erfolgt mit Hilfe von Metatags, die in XHTML oder XML unter Einsatz von RDF. Der Beschreibungssatz nach Dublin Core zeichnet sich durch Einfachheit, durch semantische Kompatibilität, durch internationale Übereinstimmung sowie durch freie Erweiterbarkeit und zukunftsweisende Kompatibilität mit RDF aus. Er liefert standardisierte semantische Informationen über Inhalte (Objekte).

Das *Ressource Description Framework* (RDF) [W3C, 2003a] erlaubt Interoperabilität zwischen verschiedenen webbasierten Anwendungen, wobei ein Austausch von Metadaten erfolgt. Primäres Entwicklungsziel des RDF war es, Semantiken für eine datengestützte Verarbeitung zur Verfügung zu stellen. Dabei ist zu berücksichtigen, dass RDF selbst keine syntaktische Konzeption für Metadaten definiert und auch nicht durch eine XML-DTD[2] definiert ist. RDF ist ausschließlich durch eine EBNF[3] definiert.

Der vom *IEEE LTSC* spezifizierte und in der Normung IEEE 1484.12 [IEEE, 2003] der LOM Working Group vorgeschlagene Standard liefert zwar umfassende Ansatzpunkte für eine systematische Strukturierung von Objekten, ist aber ausgesprochen komplex und bietet kein allgemein akzeptiertes Beschreibungsformat für multimedial ausgerichtete Elemente.

Das *Multimedia Content Description Interface* arbeitet nach dem MPEG-7[4] Standard [Fraunhofer, 2001] und wurde von der Motion Picture Expert Group 1996 ins Leben gerufen. Da weltweit immer mehr digitalisierte, audiovisuelle Informationen im W3 zur Verfügung stehen, sollte so ein allgemein akzeptiertes Beschreibungsformat für multimediale Inhalte geschaffen werden, wobei die Beschreibung mit dem Inhalt selbst assoziiert wird. Eine Aufbereitung audiovisueller Informationen nach diesem Standard ist sehr zeitaufwändig, wobei das Verhältnis von Aufwand zu Nutzen insbesondere für kurze, vertonte Videoclips nicht angemessen erscheint.

Für die strukturierte Beschreibung von Datensätzen bietet sich neben dem LOM-Standard eine weniger komplexe Standardisierung durch das W3C [W3C, 2003] an: es handelt sich hierbei um die klassifizierende Beschreibung semantischer Informationen gemäß der DCMI. Die Konventionen nach DCMI werden vom W3C vollständig unterstützt, wobei diese im RFC 2413 sowie im RFC 2731 [IETF, 1999] grundsätzlich beschrieben werden. Das IEEE LTSC spezifiziert diesen Zusammenhang [DCMI, 2002a]. Die europäische Initiative ARIADNE [ARIADNE, 2002] arbeitet z. Zt. an einer Erweite-

[1] Abgeleitet von SGML, ist XML eine Metasprache zur Definition eigener Markups, ist aber als Standard noch nicht überall implementiert.

[2] Die DTD legt fest, welche Elemente eine in diesem Dokumententyp enthaltene Auszeichnungssprache hat.

[3] Die EBNF ist eine Erweiterung der Backus-Naur Form. Sie wurde von J. Backus und P. Naur definiert, um die Programmiersprache ALGOL 60 zu beschreiben.

[4] MPEG-7 ist ein Beschreibungsformat für multimediale Inhalte, ist aber noch nicht allgemein akzeptiert.

rung der o.g. Systematik unter Einbeziehung erziehungswissenschaftlich begründeter Metadaten [ARIADNE, 2002a]. AICC und ADL sind weitere Standardisierungsinitiativen, wobei anzumerken bleibt, dass nur das IEEE die Möglichkeit hat, Vorschläge für die Etablierung eines Standards bei den zuständigen Organisationen, wie z. B. dem ANSI einzureichen.

Der LOM-Standard spielt bei der Datenrecherche im W3 eine eher untergeordnete Rolle; auch für DCMI-orientierte Webpages (ca. 0,3% aller indexierten Webpages) existieren z. Zt. noch keine indexierenden Suchmaschinen [Laurence, 1999]. Da jedoch einige Suchmaschinen konventionelle Metatags[1] ignorieren, bieten sich so Ansatzpunkte, schon existierende Lernobjekte auf der Grundlage der DCMI im W3 zu veröffentlichen, und diese so, losgelöst von etwaigen Lernumgebungen, für eine breite Öffentlichkeit zugänglich zu machen, wobei im Hinblick auf die Rechtefrage jedoch weitgehender Handlungsbedarf besteht. Zusätzlich kann generell hervorgehoben werden, "dass der Austausch, das Sharing von Learning Objects zwischen den Plattformen auf der Basis von Standards ein zentrales und in vielen Fällen auch ungelöstes Problem darstellt" [Interoperabilität, 2001]. Neben Ansätzen auf Basis des SCORM-Referenzmodells [Baudry, 2002], existieren mittlerweile auch Ansätze auf XML-Basis [Finsterle, 2002], um Austauschbarkeit und Wiederverwendbarkeit von Learning Objects zu gewährleisten.

Zur Unterstützung der DCMI existieren im W3 bereits einige Tools, um DC-konforme Datensätze zu generieren, z. B. DCdot [DCdot, 2000] von UKOLN[2], ein Template-Generator der Lund University in Helsingborg [Lund, 1998] oder auch die in der Multimediadatenbank für den Technikunterricht zur Verfügung gestellten Generatoren [MetatagsDC, 2003] oder Templates [TemplatesDC, 2003].

3.1.4 e-Learning / Blended Learning

Ein e-Learning oder Blended Learning setzt eine intensive Auseinandersetzung sowohl auf der <u>pädagogisch-prozessorientierten</u> als auch auf der <u>technisch-produktorientierten</u> Ebene voraus, um zu einer angemessenen Umsetzung der angestrebten Ziele zu gelangen.

Auf der <u>pädagogisch-prozessorientierten</u> Ebene ist der Begriff des e-Learning untrennbar mit dem Begriff des e-Teaching verbunden: beides lässt sich zur Begriffsformulierung des e-Education verschmelzen. Unter Berücksichtigung, dass hierbei jedoch nicht nur die technische Komponente eine tragende Rolle spielt, sondern auch das Individuum an sich, ist es eigentlich richtiger von einem "Hybriden Lernen" oder eben vom Blended Learning zu sprechen, d. h. ein e-Learning stellt eine methodische Variante bzw. eine ergänzende Komponente im individuellen Prozess des Lernens dar, die, entsprechend eingesetzt, das Lernen zu einem zielgerichteten Vorgang macht. Der Begriff des Selbstgesteuerten Lernens ist damit eng verbunden [Dietrich, 2000], jedoch mit Vorsicht zu betrachten, da kognitions- und experimentalpsychologische Untersuchungen gezeigt haben, dass der Lerner hierbei oft überfordert wird [Weinert, 1996; Weinert, 1997].

---

[1] Der HTML-Standard schreibt seit Version 4.0 keine konkreten Meta-Angaben mehr vor; er definiert lediglich den grundsätzlichen Aufbau einer Meta-Angabe. Nach wie vor finden sich Meta-Angaben im Header einer HTML-Datei.

[2] UKOLN: The UK Office for Library and Information Networking.

Versteht man das e-Learning als übergeordneten Begriff für softwaregestütztes Lernen, so muss nicht notwendigerweise der Computer im Mittelpunkt dieser Lernform stehen. Auf jeden Fall ist mit einem e-Learning ein Lernen möglich, das sowohl orts- als auch zeitunabhängig ist [Jakupec, 2000].

Zahlreiche Messen, Kongresse, Seminare und Workshops beschäftigen sich mit der zentralen Thematik des Einsatzes der Neuen Medien und eines damit verbundenen e-Learning in der Lehre, wobei Vertreter des Faches Technik mit sehr hohem Engagement beteiligt waren[1].

- International Conference of Scholars on Technology Education, TU Braunschweig, 24.-27.09.2000: Vorträge [Sauer, 2002; Wehling, 2002]
- PATT-11, Conference for New Media in Technology Education, Haarlem, Netherlands, 2001: Vorträge [PATT-11, 2001]
- PATT-12, Conference for Optimal Use of Resources in Technology Education, Kapstadt, Südafrika, 2001: Vorträge [PATT-12, 2002; PATT-12, 2002a]
- LIT 2002, Leipzig, 25.-28.09.2002: Vorträge [Lit, 2002]
- TERC 2002, Brisbane, Australien, 05.-07.12.2002: Vorträge [TERC, 2002]

Die Quintessenz dieser Ereignisse lässt sich in einem Satz zusammenfassen und steht momentan stellvertretend für die allgemeine Situation des e-Learning: "Beim Streit, ob bei der Entwicklung von Lerntechnologien der Technik[2] oder der Pädagogik[3] das Primat gebühre, hatten die Produzenten[4] übersehen, dass Lernen ein sozialer Prozess[5] ist, zu dem Austausch und Betreuung ebenso gehören, wie Lernziele und die Kontrolle, ob sie erreicht wurden" [Schneller, 2003]. An diesem Umstand wird auch die Einführung sog. Notebook-Universities in Duisburg oder Hannover so schnell nichts ändern, denn schon die Messe Learntec 2002 in Karlsruhe führte zu ähnlichen Ergebnissen: "Technologie ist out und Bildung in" [Reinmann-Rothmeier, 2003].

An dieser Stelle wird deutlich, dass sich hinter jeder Kommunikations- und Lernplattform ein organisations- und lerntheoretisches Modell verbirgt, das seinen Entwicklern und Nutzern nur in den wenigsten Fällen bewusst ist [Baumgartner, 2002a]. Die klassische Lehre setzt oft nur einen minimalen Aufwand voraus, während die Schaffung von Inhalten für ein e-Learning ein Expertenteam erfordert [Hübsch, 2003]. Entscheidend ist jedoch auch hier neben dem Fachwissen die didaktische Konzeption.

Besonders in der akademischen Ausbildung, die eine Mischung aus Präsenzveranstaltungen und virtueller Lehre darstellt, kann weitgehend auf eine Erstellung aufwendiger Medien verzichtet werden. Es hat sich herausgestellt, dass große Stoffmengen aus Gründen der Zeitökonomie vorwiegend über das Medium Papier akzeptiert werden,

---

[1] Aufgeführt ist hier lediglich eine kleine Auswahl. Der Autor dieser Arbeit hat an allen diesen Ereignissen aktiv teilgenommen.

[2] Mit dem Begriff Technik ist hier die technische Umsetzung gemeint.

[3] Mit dem Begriff Pädagogik sind hier didaktische, methodische, soziale und lernpsychologische Aspekte gemeint.

[4] Mit dem Begriff Produzenten sind hier die Produzenten von Lern- und Kommunikationsplattformen gemeint.

[5] Hiermit ist gemeint, dass Lernen nicht nur, sondern **auch** ein sozialer Prozess ist.

während Spezialthemen und kürzere Vorträge multimedial und interaktiv angenommen werden [Joanneum, 2002].

Die technisch-produktorientierte Ebene bezieht sich auf die technisch relevante, umsetzbare und administrative Seite des e-Learning. Neben der einfachen Frage, welche Kommunikations- und Lernplattform geeigneterweise zu wählen ist, steht die Frage der Hard- und Software im Vordergrund.

Damit sind entsprechend hohe Anforderungen verbunden:

- Die *Hardware-Seite* verlangt einen permanent zu erreichenden, stabil arbeitenden Netz-Rechner auf ausfallsicherer Basis (Redundanzsystem).
- Die *Software-Seite* erwartet ein stabil arbeitendes Betriebssystem mit effizienter Rechteverwaltung und leistungsfähigem Webserver (UNIX[1] oder Linux, Apache).

So hat sich in der Vergangenheit im Fach Technik bei der Durchführung des vorgenannten 3. Projekts gezeigt, dass eine dezentrale Installation einer Lern- und Kommunikationsplattform einer zentralen schon aus Gründen kurzer Administrationswege und schneller Eingriffsmöglichkeiten zu bevorzugen ist. Dabei sind sowohl Aspekte der Hardware bzgl. einer Ausfallsicherheit als auch der Software bzgl. einer Aktualisierung zu berücksichtigen. Hier sind gute bis sehr gute Kenntnisse im Bereich der Rechner-Hardware und des jeweils genutzten Betriebssystems sowie der gerade verwendeten Lern- und Kommunikationsplattform erforderlich. Eine derart umfassende Administration erfordert eine intensive Auseinandersetzung mit dem jeweiligen Rechnersystem und ist nur durch eine entsprechend hohe mediale Kompetenz zu leisten.

Die teilweise schon weiter oben angesprochene Betrachtung ausgewählter Lernplattformen wie Blackboard oder WebCT offenbarte den kommerziellen Hintergrund der Betreiber. Für kleine, universitäre Bereiche, die sich einem e-Learning verschreiben wollen, sind diese Produkte eher ungeeignet. Die Open-Source-Gemeinschaft bietet jedoch alternative Lernplattformen wie z. B. ILIAS oder OpenUSS [OpenUSS, 2003] an, wobei anzumerken ist, dass diese Produkte, wie die Erfahrung zeigt, oft über einen besseren und schnelleren Support bei Problemen verfügen, als ihre kommerziellen Komplementäre. Anzumerken ist, dass eine Entscheidung zugunsten des LMS ILIAS leicht fällt, da OpenUSS, wie schon erwähnt, ausschließlich in deutscher Sprache vorliegt. Außerdem orientiert sich ILIAS an Metadaten-Standards, wobei bislang die Vorgaben von ARIADNE, DCMI und IMS berücksichtigt wurden [ILIAS, 2003a].

3.2 Forderungen

Eine intensive Vermittlung von Medienkompetenz im Fach Technik kann vorwiegend im informationsumsetzenden Bereich durch computergestützte Veranstaltungen erreicht werden, womit gleichzeitig eine Verbesserung der Studierfähigkeit des Faches Technik verbunden ist. In diesem Zusammenhang ist der dadurch zu erreichende didaktische Mehrwert von wesentlicher Bedeutung.

[1] UNIX und UnixWare sind eingetragene Warenzeichen von The Open Group.

Unter Einbeziehung der weiter oben dargelegten Rahmenbedingungen hat das Fach Technik an der Universität Duisburg-Essen, Campus Essen, ein speziell auf seine Bedürfnisse zugeschnittenes *dreistufiges Modell* zum Lernen mit multimedialen Inhalten entwickelt, das sich mit drei voneinander abhängigen Stufen in diskursiver Form aufbauen lässt und durch die vorhergehenden Kapitel schon teilweise aufgezeigt wurde:

Stufe 1: Standardisierte Erstellung von multimedial orientierten Objekten [Kapitel 3.1.1 Content]

Stufe 2: Strukturiertes Ablegen von Objekten sowie neu strukturiertes Kombinieren von Objekten zu Lernobjekten (Lernmodule, Lerneinheiten, Lehrgänge) [Kapitel 3.1.2 Datenbanken und Kapitel 3.1.3 Strukturierte Datensätze]

Stufe 3: Einsatz strukturierter Lernobjekte unter Nutzung von Lern- und Kommunikationsplattformen in Präsenzveranstaltungen [Kapitel 3.1.4 e-Learning / Blended Learning].

Brauchbare Ansatzpunkte, die eine Realisierung dieses Modells möglich machen, bedürfen einer auf die organisatorischen und personellen Voraussetzungen abgestimmten Konzeption, die sich bei Bedarf an die Erfordernisse und Gegebenheiten des Faches anpassen lässt.

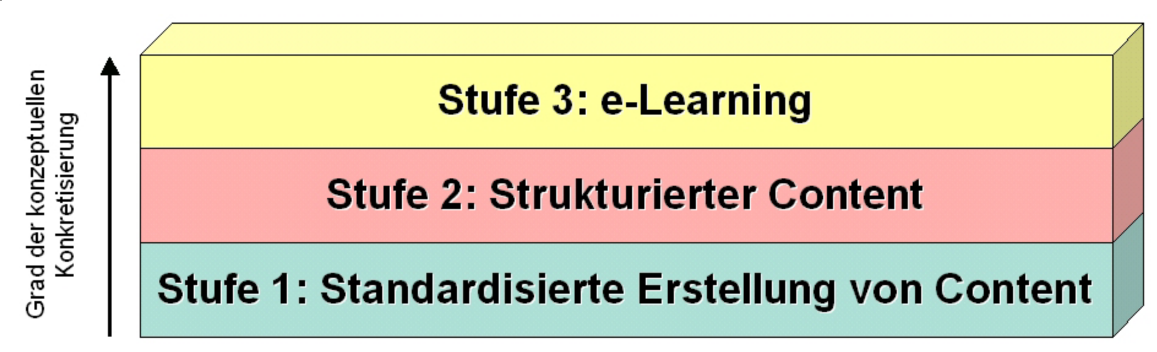


Abbildung 1: Dreistufiges Modell des Lernens mit multimedialen Inhalten

Die graphische Darstellung des dreistufigen Modells (Abbildung 1) soll durch die Angabe eines **Grades der konzeptuellen Konkretisierung** auf die hierarchische Struktur hinweisen, wodurch der Weg von der *standardisierten Erstellung von Content* über die Ablage und Kombination *strukturierten Contents* bis hin zu einem darauf aufbauenden Einsatz im *e-Learning* oder Blended Learning wiedergegeben wird[1].

Es bildet die Grundlage für alle im Folgenden aufgeführten Forderungen, die helfen sollen, ein multimediales Lernen unter Einsatz der Neuen Medien zu unterstützen.

3.2.1 Ansatzpunkt

Wie schon erwähnt orientiert sich das vorgenannte dreistufige Modell auf allen drei Stufen am Open-Source-Ansatz und berücksichtigt wesentliche Anforderungen, die zu einer hohen Akzeptanz zukünftiger Anwender beitragen (Plattformunabhängigkeit, Benut-

[1] Die mit den einzelnen Stufen verbundene Arbeitsbelastung der dieses Modell umsetzenden Mitarbeiter des Faches Technik ist durchgängig als hoch zu bezeichnen, wobei die Stufungen zeitlich betrachtet nicht nacheinander, sondern gleichzeitig umgesetzt wurden.

zerfreundlichkeit, Effizienz, Erweiterbarkeit, Datenschutz, etc.). Neben einer standardisierten Form der Entwicklung von **Objekten**[1], die den hohen Qualitätsansprüchen an wissenschaftliche Fachinhalte Rechnung trägt und die universitären Ressourcen synergetisch[2] nutzt (Stufe 1), sollte eine *datenbankbasierte Entwicklungsumgebung* zur Verfügung stehen, die es erlaubt, aus strukturiert abgelegten Objekten einfache **Module**[3] und **Module höherer Ordnung**[4] in kontrollierter und effizienter Form zu erzeugen und unter neuer Schwerpunktsetzung wieder zurück zu schreiben.

Auf diese Weise sollten sich auch Module höherer Ordnung entwickeln lassen, die aufgrund ihrer modularen Struktur flexibel handhabbar wären (Stufe 2). Eine solche *datenbankbasierte Entwicklungsumgebung* würde damit die Lücke zwischen bloßem Content (Stufe 1) und strukturierten Lernobjekten für einen Einsatz in Kommunikations- und Lernplattformen (Stufe 3) ausfüllen.

Objekte/Module

Das Objekt selbst als kleinstes und einfachstes Element muss bestimmte Anforderungen erfüllen, um zu einer Unterstützung eines multimedialen Lernens beizutragen. Unter dem Aspekt der Kombinierbarkeit einzelner Objekte zu Lernobjekten (Modulen) mit neuen Schwerpunktsetzungen müssen alle Objekte/Module in strukturierter Form geordnet und klassifiziert vorliegen.

Jedes Objekt/Modul muss daher folgende Eigenschaften besitzen. Es muss

- über einen eigenen Beschreibungssatz verfügen,
- strukturiert in einem RDBMS abgelegt sein und
- sich durch Kombinieren zu Lernobjekten (Modulen) mit neuen Schwerpunktsetzungen in modularer Form zusammensetzen lassen [Kapitel 4.1 Modularer Ansatz].

Module höherer Ordnung

Es muss die Möglichkeit bestehen, neben einfachen Modulen, die eine simple Aneinanderreihung von Objekten darstellen, auch Module höherer Ordnung zu erzeugen, die auf themenverwandte Objekte verweisen und neben technikspezifischen auch lernzielrelevante Kriterien berücksichtigen. Lernzielrelevante Kriterien sind an dieser Stelle notwendig, da ein Einsatz von Modulen höherer Ordnung in Lern- und Kommunikationsplattformen des e-Learning bzw. Blended Learning vorgesehen ist.

Daraus ergeben sich direkt weitere Eigenschaften, die erfüllt sein müssen:

- Die Kombination von zwei oder mehr Objekten erfordert bei einer Eintragung als neues **Lernmodul** auch einen neuen Datenbeschreibungssatz.

---

[1] Ein Objekt ist eine kleinste, nicht mehr weiter unterteilbare, zusammenhängende Einheit (z. B. ein Bild, ein thematisch geschlossener Text, ein Applet, eine Animation, etc.).

[2] Synergien werden im Rahmen der Objektentwicklung durch Kompetenzteams der Universität Duisburg-Essen, Campus Essen, genutzt, wobei u.a. das AVMZ eingebunden wird.

[3] Ein einfaches Modul besteht aus wenigstens zwei Objekten, die zu einer thematisch zusammenhängenden Einheit gefügt sind.

[4] Module höherer Ordnung bestehen aus wenigstens zwei einfachen Modulen, die zu einer komplexen thematischen Einheit gefügt sind.

- Die Kombination von zwei oder mehr themenverwandten Objekten erfordert bei einer Eintragung als **Lerneinheit** einen neuen Datenbeschreibungssatz sowie eine technikspezifische Beschreibung (Klassifikation).
- Die Kombination von zwei oder mehr Lerneinheiten erfordert bei einer Eintragung als **Lehrgang** einen neuen Datenbeschreibungssatz und neben einer technikspezifischen zusätzlich eine fachdidaktische Beschreibung (Lernziele).

Allgemeine Eigenschaften

Neben den o.g. Anforderungen für Objekte/Module und Module höherer Ordnung sind zusätzlich allgemeine Anforderungen zu stellen, die sowohl für Objekte als auch für Module, ganz gleich welcher Art, gelten müssen.

Sie müssen

- fachwissenschaftlich einwandfrei sein,
- bewertbar sein,
- jedem registriertem Nutzer orts- und zeitunabhängig zugänglich sein,
- Eigenschaften der Interoperabilität erfüllen und
- portierbar sein.

Eine *datenbankbasierte Entwicklungsumgebung für eine Generierung von Lernmodulen, Lerneinheiten und Lehrgängen* stellt eine grundsätzliche Lösung des oben beschriebenen Ansatzpunktes dar. Nach Fertigstellung ist sie im Seminareinsatz evaluativ zu prüfen und anschließend gegebenenfalls zu modifizieren. Die Sicherung der Nachhaltigkeit sollte durch geeignetes Personal (Hilfskräfte) erfolgen.

3.2.2 Realisierungsmöglichkeit

Die im folgenden vorgestellte Realisierungsmöglichkeit stellt über die Umsetzung des **konzeptionellen Ansatzes** unter Integration von **Open-Source-Software** daraus resultierende **Umsetzungskriterien** für die Unterstützung eines multimedialen Lernens unter Einsatz der Neuen Medien auf. Ihr angeschlossen ist eine Betrachtung der daraus resultierenden Umsetzungskonsequenzen, dargestellt am Beispiel des L.A.M.P.–Ansatzes unter Hinweis auf einzusetzende Hardware.

Konzeptioneller Ansatz

Ausgehend von den Projekten 1 bis 3 [Kapitel 2.3 Fachspezifisch] und dem dreistufigen Modell des Lernens mit multimedialen Inhalten (Stufe 1 bis Stufe 3) [Kapitel 3.2 Forderungen] bietet sich der folgende Weg für ein strukturiertes Ablegen von Objekten sowie ein neu strukturiertes Kombinieren derselben zu Lernobjekten an:

bereits existierende Inhalte (Content von Projekt 1) und standardisierte Inhalte (Projekt 2, Stufe 1)[1] *werden strukturiert abgelegt (Projekt 2, Stufe 2) und anschließend zu*

[1] Durch Projekt 2 werden die Stufen 1 und 2 abgedeckt.

Evaluationszwecken in Lern- und Kommunikationsplattformen eingesetzt (Projekt 3, Stufe 3)[1].

Der abschließende Einsatz einer Kommunikations- und Lernplattform (e-Learning) komplettiert die Konzeption des dreistufigen Modells zum Lernen unter Einsatz der Neuen Medien (Abbildung 2).

Im Folgenden wird erläutert, wie unter Einbeziehung des vorhergehenden Kapitels 3.2.1 das Projekt 2 auf der Stufe 2 einen begründeten Rahmen für eine mögliche Realisierung liefern kann.

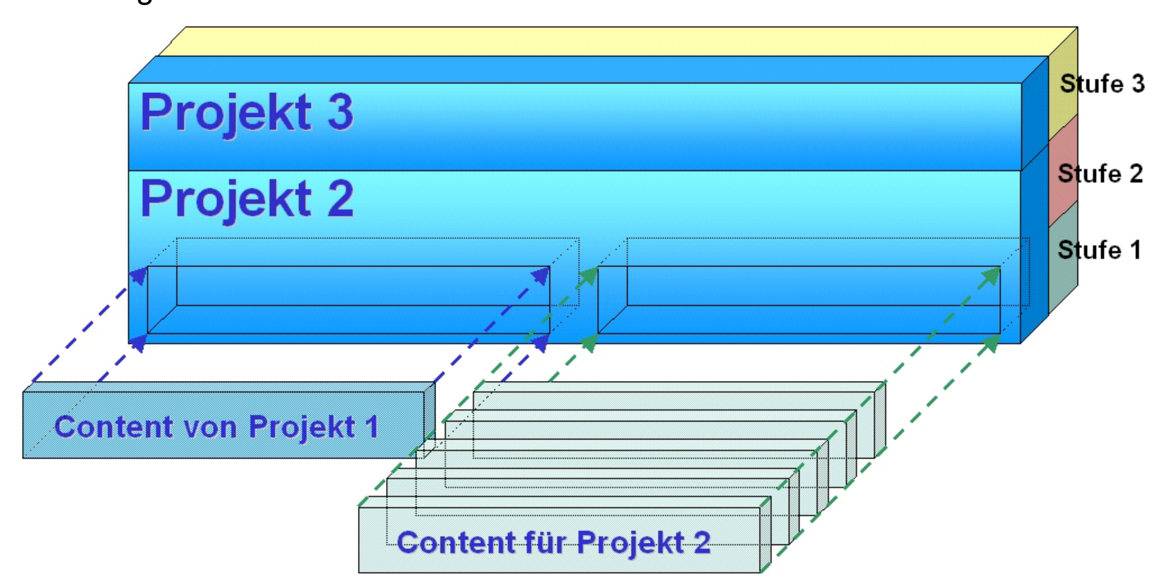


Abbildung 2: Verknüpfung von Konzeption und Einzelprojekten

Die Interdependenzen von Konzeption (dreistufiges Modell, im Hintergrund) und einzelnen Projekten bzw. Inhalten (Content von Projekt 1, Projekte 2 und 3) sind in Abbildung 2 wiedergegeben:

Das schon abgeschlossene Projekt 1 lieferte, bezogen auf die Projektinhalte, modularisierten Content in einer standardisierten Form. Dieser **Content von Projekt 1** wird, entsprechend aufbereitet, für eine Integration in **Projekt 2** zur Verfügung gestellt. Die integrationsfähigen Inhalte von Projekt 1 eignen sich sehr gut für eine Übertragung auf Stufe 2 (strukturierter Content). Zusätzlich wird durch die Produktion von modularem **Content für Projekt 2** das in der Projektbeschreibung formulierte Thema [Kapitel 2.3 Fachspezifisch] erfüllt. Somit sind die beiden Stufen 1 (standardisierte Erstellung von Content) und 2 (strukturierter Content) des dreistufigen Modells durch Projekt 2 sinnvoll miteinander verbunden. Schließlich setzt **Projekt 3** auf die durch Projekt 2 erreichten Ergebnisse auf und komplettiert das dreistufige Modell durch den Einsatz von Kommunikations- und Lernplattformen. Wie schon erwähnt wurden die hier genannten Projekte vorwiegend gleichzeitig und nicht nacheinander durchgeführt!

[1] Die Ergebnisse von Projekt 2 lassen sich direkt durch Projekt 3 nutzen.

Open-Source-Software

Die Verwendung von Open-Source-Software bietet eine kostenneutrale Umsetzung von Stufe 2 des vorgenannten dreistufigen Modells des Lernens mit multimedialen Inhalten [Kapitel 3.2 Forderungen]. So wird sichergestellt, dass jeder Nutzer unabhängig von seinem jeweils verwendeten Betriebssystem mit den netzgebundenen Programmen arbeiten kann.

Eine sinnvolle Kombination von Open-Source-Software stellt der Ansatz nach L.A.M.P. dar, der in Abbildung 3 visualisiert ist [Goller, 2002].

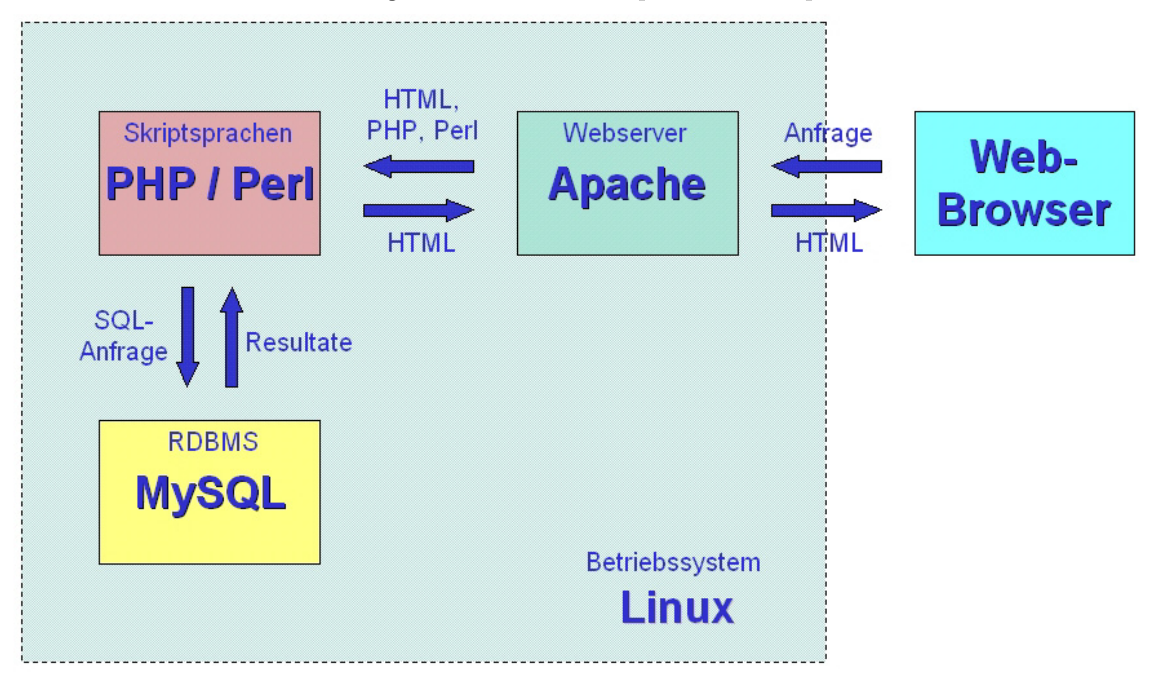


Abbildung 3: Visualisierung des L.A.M.P.-Ansatzes

Hier wird deutlich, dass der Webserver Apache das zentrale Bindeglied zwischen Nutzer und Server[1] darstellt. Er regelt den Datenaustausch mit dem RDBMS durch Einsatz der Skriptsprachen PHP / Perl. **Apache**, **MySQL** und **PHP** / **Perl** laufen unter dem Betriebssystem **Linux**.

Die Wahl von MySQL als RDBMS impliziert gleichzeitig, wie oben bereits erwähnt, die Verwendung von PHP und Perl als serverseitige Programmiersprachen zur dynamischen Generierung von Web-Seiten bzw. zur Implementierung administrativer Vorgänge. Da sowohl MySQL als auch PHP und Perl freie Software sind und eine breite Unterstützung innerhalb der Gemeinschaft der Linux-Anwender finden, besteht der nächste konsequente Schritt darin, ein UNIX-basiertes Betriebssystem oder eines der vielen UNIX-Derivate einzusetzen. Die meisten UNIX-Derivate, wie Debian, Red Hat oder Suse-Linux stehen ebenfalls kostenlos als Open-Source-Produkte zur Verfügung.

[1] Mit Server ist in diesem Fall die konkrete Hardware, also der Rechner selbst, gemeint. Der Webserver Apache ist ein reines Softwareprodukt.

Es bleibt anzumerken, dass MySQL auch für Windows-basierte Betriebssysteme verfügbar ist; ebenso PHP und auch der Apache-Webserver. Es würde also nichts dagegen sprechen, beispielsweise einen Ansatz nach W.A.M.P. zu verfolgen; leider existiert zu dieser Thematik kaum Literatur. Da Windows-basierte Betriebssysteme kommerziell vertrieben werden, würde dieser Umstand außerdem einer konsequenten Umsetzung des Open-Source-Ansatzes widersprechen. Es sei kurz angemerkt, dass es neben PHP und Perl alternative Möglichkeiten wie ASP [Falz, 2002] und ASP.Net [Lorenz, 2002] von Microsoft, ColdFusion [Schmitz, 2003] von Macromedia oder VelociGen [Blue, 2003] von VelociGen Inc.[1] gibt. Sie sind jedoch nicht unter allen relevanten Betriebssystemen einsetzbar, bzw. verfügen nur über einen geringen Sprachumfang, sind propriertär ausgerichtet oder werden durch hohe Lizenzgebühren indiskutabel.

Umsetzungskriterien

Die über diesen Ansatz zu realisierende *datenbankbasierte Entwicklungsumgebung* muss dabei sämtliche Kriterien erfüllen, die für einen effektiven Umgang auf netzgebundener Ebene notwendig sind. Neben den Aspekten der Interoperabilität und Portabilität sind das:

- Browserunabhängigkeit
- Ausgewogenes Webdesign[2] durch dynamisches (X)HTML
- Bilingualität
- Weitgehende Vermeidung unzureichend etablierter Standards
- Weitgehende Vermeidung proprietärer Standards
- Ladezeitoptimierte Inhalte
- Einsatz interaktiver Elemente
- Integrierte Hilfefunktion
- Einfache e-Mail-Dienste
- Schnelle und zuverlässige Internetanbindung
- Ausfallsicherheit durch Redundanzhardware.

So werden durch Einhaltung dieser Kriterien bei der Realisierung der *datenbankbasierten Entwicklungsumgebung* jedem Nutzer einfache Schnittstellen[3] angeboten, um digitalisierte Inhalte in den wichtigsten, netzgestützten Formaten strukturiert abzulegen, neu anzuordnen und allen weiteren Anwendern unter speziellen Schwerpunktsetzungen zur Verfügung zu stellen. Dabei sollten keine besonderen informationstechnischen Spezialkenntnisse notwendig sein.

Neben den o.g. Kriterien zur Realisierungsmöglichkeit ist anzumerken, dass eine fast unüberschaubare Vielfalt an Hilfsmitteln zur Unterstützung bei der Systementwicklung

[1] VelociGen ist das eingetragene Warenzeichen von Blue Titan Software.

[2] Webdesign heißt hier: Informationen für das Internet nutzerfreundlich aufbereiten, gliedern, sinnvoll miteinander verknüpfen und durch eine ausgewogene Benutzerführung auf einfache Weise zugänglich machen.

[3] Mit Schnittstellen ist hier die Kommunikation Mensch/Maschine über ein GUI gemeint.

angeboten werden, diese aber mit teilweise sehr hohen Lizenzgebühren verbunden sind oder unbefriedigend arbeiten[1]. Insbesondere unter Berücksichtigung der Browserunabhängigkeit bleibt nichts anderes übrig, als (X)HTML-Code von Hand zu setzen und bei kritisch erscheinenden Konstrukten durch Einsatz der verschiedenen Browser in unterschiedlichen Versionen die beabsichtigte Unabhängigkeit durch das Ergebnis zu prüfen.

Des weiteren ist zu beachten, dass von einer Verwendung noch nicht etablierter und implementierter Standards, wie z. B. XML, vorerst abgesehen werden sollte, da insbesondere ältere Browser noch nicht mit Spezifikationen dieser Art umgehen können; das gleiche gilt für proprietäre Standards, da der Eigentümer jederzeit systemrelevante Detailzusammenhänge ändern kann. XHTML in der Version 1.0 und HTML in der Version 4.0 haben sich längst bewährt, sind standardisiert und werden auch in Zukunft durch jeden Webbrowser interpretierbar sein. XHTML lässt sich als ein XML-gerechtes HTML verstehen und liegt seit Januar 2000 in der Version 1.0 als Empfehlung des W3C vor. Es hat damit den gleichen verbindlichen Stellenwert wie HTML in der Version 4.0 [W3C, 2003b].

Die Einbindung ladezeitoptimierter Graphiken muss genauso selbstverständlich sein wie das Einbringen interaktiver Elemente in den Seitenaufbau: beides macht ein nutzerfreundlichen Webdesign aus und erleichtert ein netzgebundenes Arbeiten wesentlich.

Erst der zusätzliche Einsatz von **Javascript** zur Implementierung von Online-Hilfefunktionen liefert ein durchweg dynamisches Webdesign [Flanagan, 2002]. In diesem Zusammenhang ist zu beachten, dass auf administrativer Ebene die Programmiersprache Perl schon aus traditionell bewährten Gründen der Sprache PHP vorzuziehen ist; so wiesen ältere Versionen von PHP (bis Version 3) z. B. Probleme im sicheren Umgang mit Benutzereingaben, mit der Ausgabepufferung von Daten und mit nutzerdefinierten Sessions auf [phpResource, 2003].

Neben diesen softwareseitig angesiedelten Kriterien sind die hardwareseitig zu erfüllenden Kriterien mindestens genauso wichtig: eine unzuverlässige und langsame Internetanbindung ist ebenso unerwünscht, wie der ständige Ausfall des Host-Rechners. Diese Kriterien lassen sich nur durch die Wahl eines guten und preisgünstigen ISP und durch den Einsatz von Redundanzsystemen, wie z. B. Replikationssystemen[2], RAID-Systemen[3] oder sogar Clusterung[4] zur Realisierung eines Load Balancing[5], erfüllen. Abschließend ist über eine Trennung von Webserver und Datenserver nachzudenken, die natürlich auch eine aufwendigere Administration des Gesamtsystems nach sich zieht.

Neben dem Vorteil der niedrigen Kosten existieren durch diese Realisierungsmöglichkeit allerdings auch gravierende Nachteile.

---

[1] Z. B.: Nusphere, phped, perledit, Frontpage, Golive.

[2] Synchronisation des Datenbestandes durch ein Master/Slave-System.

[3] Festplatten-Spiegelung (Mirroring) im Level RAID 1 oder RAID 10.

[4] Speicherung gemeinsam benutzter Datensätze in gleichen oder physikalisch benachbarten Segmenten.

[5] Automatische Umverteilung von Anfragen auf verschiedene Rechner.

Nachteilig wirkt sich z. B. aus, dass sie nur durch eine sehr intensive Beschäftigung mit der damit verbundenen Materie durchzuführen ist. Der Einsatz von Programmen aus dem Open-Source-Bereich erfordert in aller Regel eine hohe Einarbeitungszeit, da viele Sachzusammenhänge von den jeweiligen Autoren als 'bekannt' vorausgesetzt werden und dementsprechend nicht oder nur ungenügend dokumentiert sind. Im Hinblick auf eine Realisierung nach dem L.A.M.P.-Ansatz seien hier einzelne Problematiken exemplarisch angesprochen:

- **Linux:** Auf der Ebene des Betriebssystems soll ein UNIX-Derivat, wie Debian, Red Hat, OpenLinux Caldera oder SuSE-Linux[1] zum Einsatz kommen. Alle diese Systeme haben Stärken und Schwächen auf den unterschiedlichsten Gebieten. So bietet beispielsweise SuSE-Linux einen exzellenten Online-Support und umfangreiche deutsch- und englischsprachige Literatur für ein Arbeiten mit diesem Betriebssystem, während Red Hat eine ähnliche Unterstützung ausschließlich auf den englischsprachigen Raum beschränkt. Unter zusätzlicher Berücksichtigung der Hardwareseite ist es von entscheidender Bedeutung, welcher Hardware-Anbieter für RAID-Systeme nicht nur für SCO-UNIX[2], sondern auch für SuSE-Linux einen zuverlässigen Treiber-Support anbieten. So liefert Promise, einer der etabliertesten Hersteller von RAID-Systemen auf ATA IDE-Basis lediglich SuSE-Treiber für die Versionen 7.1 und 7.3 (z. Zt. ist die Version 8.1 aktuell), die sich leider noch im Beta-Stadium der Entwicklung befinden und daher instabil arbeiten, für Red Hat hingegen existiert ein vollwertiger Treiber-Support [Promise, 2003].
- **Apache:** Der Apache als ein reines Open-Source-Produkt ist der am häufigsten im W3 eingesetzte Webserver; er arbeitet ausgesprochen stabil und wird laufend verbessert sowie modifiziert. Z. Zt. existieren zwei Versionen, die neuere Version 2.0.44 sowie die ältere Version 1.3.27, wobei die ältere Version nicht mehr weiter entwickelt wird. Leider sind die Rahmenbedingungen, unter denen ein solcher Webserver in ein Linux-System integriert wird, sowohl von der Version des zu nutzenden Linux-Systems, als auch von der Version des Apache Webservers selbst abhängig. D. h.: die Dateipfade zum Webserver sind immer andere und auch die Existenz und Namensgebung seiner Konfigurationsdateien sind versionsabhängig. Ein Versionsupdate erfordert somit eine gewisse Einarbeitungszeit. Kommerzielle Produkte wie beispielsweise der Enterprise-Server von Netscape oder der IIS von Microsoft sind wesentlich benutzerfreundlicher.
- **MySQL:** Um effektiv mit MySQL arbeiten zu können, sind genaue Kenntnisse des RDBMS im Allgemeinen und im Speziellen erforderlich. Der prinzipielle Aufbau servergebundener RDBMS und die serverseitige Implementierung einer MySQL-Datenbank erfordern eine umfassende Kenntnis der ineinander greifenden Prozesse, die auf der Ebene des Betriebssystems ablaufen. Neben der Planung des aufzusetzenden RDBMS müssen Kenntnisse über Begriffe wie Datenbankprogrammierung, Datenbankrecherche und Datenmanipulation als elementare Planungsgrund-

[1] Debian, Red Hat, Caldera und SuSE sind eingetragene Warenzeichen der durch das Warenzeichen selbst benannten Firmen.

[2] SCO ist ein eingetragenes Warenzeichen von Caldera International Inc.

lagen vorhanden sein, um bei einigen grundlegenden Begriffen zu bleiben [Bilke, 2001].

- **Perl:** Diese Skriptsprache ist eine sehr alte und bewährte Sprache von außerordentlicher Komplexität. Durch den Einsatz von Perl, als eine Möglichkeit der CGI-Programmierung[1] lassen sich alle softwarebedingten, serverseitigen Probleme lösen. Ihre Vielfalt erfordert allerdings immer wieder eine intensive Auseinandersetzung mit ihrer (gewöhnungsbedürftigen) Syntax. So tauchen beispielsweise Reguläre Ausdrücke[2] in allen Perl-Skripten auf, in denen es um eine Bearbeitung von textuellen Elementen geht. Für sich genommen stellen diese fast kryptisch anmutenden Programmcodes schon fast eine eigene Fachwissenschaft dar [Friedl, 2001]. Eine Verarbeitung solcher Elemente ist im Zusammenhang mit Datenbankrecherchen im Hinblick auf eine Implementierung effektiver Suchalgorithmen jedoch unumgänglich.
- **PHP:** Neben tiefgreifenden Kenntnissen der Programmiersprache Perl ist eine Anwendung von dynamischem (X)HTML auf der Basis von PHP erforderlich. Es ist relativ einfach, PHP-Skripte zu erstellen, die simple Operationen ausführen; wesentlich anspruchsvoller sind Skripte, die zeitgleiche Datenbank-Anfragen verarbeiten oder einen Webserver auf Betriebssystemebene ansprechen [Enseleit, 2001]. Die effektive Anwendung dieser Skript-Sprache setzt daher ein genaues Verständnis der Funktionsweise des gewählten Betriebssystems und des darauf installierten Webservers voraus; das gleiche gilt natürlich auch für die Skript-Sprache Perl. Es erübrigt sich fast zu erwähnen, dass intensive Kenntnisse über Netzwerke sowie deren Topologien und Protokolle (OSI-Schichtenmodell[3]) ebenso vorauszusetzen sind, wie solche die Auszeichnungssprachen HTML und XHTML betreffend.

Fasst man die Realisierungsmöglichkeiten zusammen, so ergibt sich schnell folgendes Bild: Für eine konsequente Umsetzung des L.A.M.P.-Ansatzes ist ein ausgesprochen hohes Maß an Fachwissen und medialer Kompetenz erforderlich. Hier wird eine umfassende informationstechnische Bildung verlangt, die nicht nur ein intensives Spezialwissen auf den einzelnen o.g. Gebieten voraussetzt, sondern daneben ein zusätzliches, systemübergreifendes und systemverbindendes Expertenwissen verbunden mit langjähriger Erfahrung.

3.2.3 Zielsetzung

Von einer Analyse der zur Verfügung stehenden Ressourcen bis hin zum gezielten Einsatz einer *datenbankbasierten Entwicklungsumgebung* in Seminaren, Vorlesungen, Übungen und zum autarken Lernen werden die zu erwartenden Umsetzungsergebnisse im folgenden schrittweise formuliert und anschließend ausführlich erläutert. Nur so kann eine Umsetzung der im vorhergehenden Kapitel formulierten Forderungen erfolgen. Unter Berücksichtigung der dort genannten Zusammenhänge können, bezogen sowohl auf

---

[1] Das CGI lässt sich z. B. auch über C++ programmieren.

[2] Reguläre Ausdrücke stellen eine generelle Notation zur Beschreibung von Textmustern sowie zum Prüfen und Manipulieren von Texten dar.

[3] Bei der Datenübertragung werden die genutzten Protokolle in Schichten entsprechend dem von der ISO genormten OSI-Referenzmodell verwendet.

die Entwicklung, als auch auf die Fertigstellung und den darauf folgenden Einsatz der *datenbankbasierten Entwicklungsumgebung*, Umsetzungsergebnisse erwartet werden, die mit folgenden wesentlichen Fragen verbunden sind:

a) Welche Ergebnisse liefert eine Analyse der zur Verfügung stehenden Ressourcen?
b) Welche Vorbereitung einer programmtechnischen Realisierung sind sowohl auf der Hardware- als auch auf der Softwareseite zu treffen?
c) Wie lässt sich eine programmtechnische Realisierung konkret umsetzen?
d) Welche wesentlichen sicherheitstechnischen Aspekte sind zu berücksichtigen?
e) Welchen Beitrag kann die *datenbankbasierte Entwicklungsumgebung* zur informationstechnischen Weiterbildung im Sinne eines didaktischen Mehrwertes im Fach Technik leisten?
f) Wann, wo und wie wird die *datenbankbasierte Entwicklungsumgebung* letztlich eingesetzt?

Die o.g. Punkte werden im folgenden näher ausgeführt.

Punkt a): Am Anfang einer jeden umfangreichen Aufgabe steht eine Analyse der zur Verfügung stehenden Ressourcen. Darunter fällt neben dem eigenen Engagement auch eine Einbeziehung möglicher zusätzlicher Potentiale, kurz: welche Manpower steht über welchem Zeitraum zur Verfügung? Diese Fragestellung wurde im Abschluss des vorhergehenden Kapitels indirekt beantwortet: aufgrund der vorauszusetzenden, umfassenden informationstechnischen Bildung sowie der erforderlichen vertieften, speziellen Kenntnisse auf den Gebieten der Datenbank-, Webseiten- und Webserverprogrammierung auf Linux-Ebene steht keine zusätzliche Manpower zur Verfügung.

Punkt (b): Eine unzuverlässige und langsame Internetanbindung ist ebenso unerwünscht, wie der ständige Ausfall des Host-Rechners, d. h. neben einem global erreichbarem und verlässlichem Netzwerk muss rechnerseitig ein zuverlässiges Redundanzsystem zur Verfügung stehen. Diese Kriterien lassen sich nur durch die Wahl eines guten und preisgünstigen ISP und durch den Einsatz von Redundanzsystemen erfüllen. Ein solches System muss ebenfalls in der Lage sein, Spannungsschwankungen und Stromausfälle durch Einsatz einer USV auszugleichen bzw. über einen angemessenen Zeitraum zu überbrücken. Schließlich ist über eine Trennung von Webserver und Datenserver nachzudenken, die natürlich auch eine aufwendigere Administration des Gesamtsystems nach sich zieht. Unter Berücksichtigung des o.g. Ansatzes nach dem Open-Source-Modell muss das System zusätzlich über ein ausgesprochen stabil arbeitendes, netzwerkfähiges Betriebssystem verfügen. Dabei muss dieses Betriebssystem die Möglichkeit bieten, den vorgenannten Ansatz nach L.A.M.P. zu implementieren, also zumindest ein kostenloses UNIX-Derivat, wie z. B. Linux sein.

Punkt (c): Eine detaillierte Umsetzung der programmtechnischen Realisierung nach dem L.A.M.P.-Ansatz erfordert parallel zum Einsatz des Apache Webservers sowie der Programmiersprachen PHP und Perl die Implementierung von MySQL als RDBMS. Die Architektur der zu entwickelnden Datenbank ist bis höchstens zur 3. Normalform[1] zu

[1] Verschiedene Stufungen in der Optimierung eines Datenmodells werden als Normalformen bezeichnet.

erstellen, um später noch erweiterte Datensätze hinzufügen zu können. Falls notwendig, sollte die Architektur der Datenbank später im Hinblick auf höhere Normalformen endgültig optimiert werden.

Bei File-generierenden, Datenbank-schreibenden und Rechte-vergebenden Prozessen innerhalb des Datenbank- und Filesystems sollte Perl, bei Datenbank-abfragenden und Ergebnis-darstellenden Vorgängen PHP verwendet werden. Bei unkritischen Prozessen sind auch Schnittmengen möglich: so lassen sich unter PHP temporäre Dateien generieren, die später durch ein Perl-Skript in die Datenbank eingetragen werden können; u.U. lässt sich auch Perl für eine Generierung dynamischer Webseiten verwenden.

Auf einem Linux-basierten Betriebssystem wird der Apache als Webserver installiert, um den netzgebundenen Datenaustausch zu regeln. Zusätzlich sollte wenigstens ein e-Mail-Dienst aufgesetzt sein, um potentiellen Nutzern der *datenbankbasierten Entwicklungsumgebung* Rückmeldungen durchgeführter Aktionen zu übermitteln.

Punkt (d): Für die Implementierung sicherheitsrelevanter Instrumentarien sind allgemein übliche Vorkehrungen zu treffen. Darunter fällt hardwareseitig das Schließen sämtlicher nicht benötigter Ports[1]. Softwareseitig werden in alle eingaberelevanten Programme spezielle Fehlerabfangroutinen eingebaut, die vor unbefugtem Zugriff und Missbrauch schützen sollen. Zusätzlich werden sämtliche netzgebundenen Vorgänge durch Access Log Files[2] gespeichert und in regelmäßigen Abständen einer Auswertung unterzogen.

Bis zu ihrer globalen Freischaltung erhält die *datenbankbasierte Entwicklungsumgebung* einen allgemeinen Zugangsschutz, der nur bekannten Gästen und registrierten Nutzern Zugang gewährt. Ein zweiter, spezieller Zugangsschutz stellt sicher, dass ausschließlich registrierte Nutzer Objekte/Module in die Datenbank eintragen können. Der allgemeine Authentifizierungsprozess wird durch den Webserver der Spezielle durch die Datenbank selbst vorgenommen.

An dieser Stelle sei angemerkt, dass eine vollkommene Sicherheit in einem netzgebundenen, datenverarbeitenden System nicht zu leisten ist, wobei ein Linux-basiertes System schon aufgrund seiner standardisierten Sicherheitsmechanismen weniger angreifbar ist als ein Windows-basiertes. Wer jedoch über genügend kriminelle Energie verfügt, kann jedes noch so gut geschützte System aushebeln. Es muss daher reichen - auch in Anbetracht der zur Verfügung stehenden Manpower - lediglich die wichtigsten Sicherheitsvorkehrungen zu treffen.

Punkt (e): Die *datenbankbasierte Entwicklungsumgebung* ist eine Schnittstelle für alle diejenigen, die am Fach Technik interessiert sind. Unter rein fachwissenschaftlichem Aspekt bietet sie ihren Nutzern (hier sind Dozenten, Ingenieure, Techniker, etc. angesprochen) die Möglichkeit, ihre jeweiligen speziellen Themen unter Nutzung technikspezifischer Kriterien einzuordnen, zu klassifizieren und so einer geschlossenen Community zur Verfügung zu stellen. Unter einem gemischten fachwissenschaftlichem und fachdidaktischem Aspekt eröffnet sie ihren Nutzern (hier sind Dozenten, Lehrer, Ausbil-

---

[1] Über Ports werden spezielle Rechnerdienste und Protokolle angesprochen: z. B.: ftp, telnet, www.

[2] Bei jedem Zugriff auf einen Webserver werden Servername / IP-Adresse, Datum, Zeit, Datei und Ergebnis des Zugriffs in einem speziellen File protokolliert.

der, Referendare, etc. angesprochen) die Möglichkeit, neben einer Einordnung technikspezifischer Kriterien zusätzlich lernzielrelevante Eintragungen vorzunehmen. Sie stellt somit neben ihrer Eigenschaft, Daten strukturiert zu sammeln und neu zu kombinieren, ein innovatives Instrumentarium der Planung und Generierung sowohl von rein fachwissenschaftlich fundierten Lernmodulen bzw. Lerneinheiten, als auch von fachdidaktisch begründeten Lehrgängen dar.

Des weiteren sollte innerhalb einer Community die Möglichkeit bestehen, sämtliche durch die *datenbankbasierte Entwicklungsumgebung* verfügbaren Objekte und Module zu bewerten, wodurch jeder aktive Nutzer gehalten ist, seine eigenen Objekte und Module vorher eingehend auf sachliche und fachliche Richtigkeit zu prüfen.

Die angestrebte Bilingualität macht die *datenbankbasierte Entwicklungsumgebung* zu einem nachhaltigen System für einen länderübergreifenden Technikunterricht, wobei neben der jetzt schon erfolgreichen Durchführung bundeslandverbindender Projekte zunehmend auch eine Basis für die Durchführung europaweiter Projekte geschaffen wird.

Somit kann die *datenbankbasierte Entwicklungsumgebung* einen wesentlichen Beitrag zum didaktischen Mehrwert durch fundierte fachwissenschaftliche, fachdidaktische und informationstechnische Weiterbildung liefern und so Ansatzpunkte zu einer erheblichen Verbesserung der Studierfähigkeit des Faches Technik sowie zu einer gestärkten Position dieses Faches im bildungspolitischen Umfeld bieten.

Punkt (f): Nach Abschluss der Arbeiten an den Projekten 2 und 3 wird auch die 3. und damit letzte Stufe des vorgenannten Modells für ein Lernen mit multimedialen Inhalten erreicht sein. Die *datenbankbasierte Entwicklungsumgebung* wird bis dahin vollständig fertig gestellt und betriebsbereit sein. Vorgesehen ist ein kontinuierlicher Einsatz in den Veranstaltungen des Faches Technik, wobei in regelmäßigen Abständen immer wieder Verbesserungen und Versions-Updates erfolgen sollen. Parallel dazu wird sie auf Messen und bei Workshops vorgestellt und eingesetzt. Letztlich wird sie einer globalen Community in bilingualer Form zur Verfügung stehen, um u.a. einen europaweiten Austausch technikrelevanter Inhalte zu initiieren.

4 Realisierung

Dem Open-Source-Ansatz folgend wurde mit Beginn des 2. Projekts [Kapitel 2.3 Fachspezifisch] ein Webserver mit der IP-Adresse 132.252.134.26 auf Linux-Basis eingerichtet und an das Hochschuldatennetz der Universität Duisburg-Essen, Campus Essen, angeschlossen. Tabelle 1 gibt einen Überblick über die wichtigsten vorgenommenen Modifikationen der Hard- und Software vom Beginn des 2. Projekts an bis zum jetzigen Zeitpunkt.

	Beginn des 2. Projekts	**Letzte Modifizierung**
CPU:	Pentium II/300 Prozessor[1]	Pentium II/300 Prozessor
RAM:	128 MB	256 MB
HDs:	2 x 7 GB	2 x 80 GB
Controller:	SR2200[2] ATA RAID Level 1	SR2200 ATA RAID Level 1
Netzwerkkarte:	3COM 3C595-TX	Accton 1207-TX
Linux-Kernel:	Version 2.4.0	Version 2.4.19
MySQL:	Version 3.23.30 gamma	Version 4.0.12
Apache:	Version 1.3.14	Version 2.0.44
PHP:	Version 4.0.4p11	Version 4.3.1
Perl:	Version 5.6.0	Version 5.8.0
PhpMyAdmin:	Version 2.1.0	Version 2.4.0

Tabelle 1: Modifikationen der Hard- und Software

4.1 Modularer Ansatz

Im Fach Technik wurde schon früh ein modularer Ansatz für die Unterstützung eines multimedialen Lernens unter Einsatz der Neuen Medien durch die Entwicklung komponentenbasierter Lernsoftware im Rahmen des 1. Projekts [Kapitel 2.3 Fachspezifisch] umgesetzt. Ausgehend von einzelnen Komponenten wurde zum damaligen Zeitpunkt ein Modul zum Thema Lichtwellenleiter entwickelt [Asano, 2001]. Zu diesem Zweck wurden drei wesentliche Komponenten (*Geschichte der optischen Nachrichtentechnik*, *Grundlagen der Nachrichtentechnik* und *Physikalische und technische Grundlagen von Lichtwellenleitern*) geschaffen, die später zu einem einheitlichen MultiMediaModul[3] zu-

[1] Dieser Prozessortyp ist für seine hohe Ausfallsicherheit bekannt. Pentium II/300 Prozessoren arbeiten auch nach einem Ausfall des Prozessorlüfters problemlos weiter.

[2] Bei diesem System handelt es sich um einen microcontroller-gesteuerten Hardwaresatz der Firma Sonic.

[3] Der Begriff des Moduls bezieht sich in diesem Fall nicht auf die in dieser Arbeit benannten Module oder Lernobjekte, sondern betrifft den durch die Entwicklungsarbeit des MultiMediaModuls Lichtwellenleiter bedingten modularen Aufbau.

sammengefasst wurden.

Dieser **komponentenorientierte Ansatz** folgte der modularen Systematik, einzelne Bausteine getrennt voneinander zu entwickeln und später zu einer fest gefügten, übergeordneten Thematik zu verbinden. So setzt sich beispielsweise die Komponente *Geschichte der optischen Nachrichtentechnik* aus den einzeln entwickelten Bausteinen *Der Fackeltelegraph von Polybios*, *Der Balkentelegraph von Chappe* und *Der Klappentelegraph von Murray*[1] zusammen.

Leider hatte dieser komponentenorientierte Ansatz den Nachteil geringer Flexibilität, da die wesentlichen Komponenten in starrer, vorgegebener Form vorlagen. Es war daher nicht möglich, einzelne Aspekte einer speziellen Thematik in einem Gesamtzusammenhang zu betrachten, weil die dafür benötigten Elemente weit über das Modul verstreut vorlagen. Diese Problematik gab die künftige Arbeitsrichtung vor, um einzelne Komponenten für ein flexibles Arbeiten kombinieren zu können: elementare Bausteine sollten Bestandteile eines Baukastens werden, aus dem komplexere Komponenten durch freie Kombination in modularer Form [Kapitel 3.2.1 Ansatzpunkt] mit neuen Schwerpunktsetzungen [Kapitel 3 Problemstellung] zusammengesetzt werden konnten.

Dieser Ansatz wird im Fach Technik als **modularer Ansatz** bezeichnet und ist in Analogie mit dem gerade Beschriebenen durch die in dieser Arbeit verwendeten Begriffsformulierungen in folgender Weise verbunden: *elementare Bausteine* werden als **Objekte** bezeichnet, während die *Bestandteile eines Baukastens* in einer Datenbank vorliegen und *komplexere Komponenten in modularer Form* **Lernobjekte** genannt werden.

Dieser modulare Ansatz hat die Vorteile, dass:

- einmal formulierte Objekte nicht immer wieder neu in anderen Zusammenhängen erstellt werden müssen, sondern jederzeit über eine Datenbank zur Verfügung stehen,
- alternative Thematiken leicht gegeneinander austauschbar sind,
- die Möglichkeit eines neu strukturierten Kombinierens von Objekten zu Lernobjekten besteht und
- die Transparenz, Pflege und Wiederverwendbarkeit von Objekten und Lernobjekten erhöht wird.

Da hier sowohl von Objekten, als auch von Lernobjekten die Rede ist, erscheint es sinnvoll, diese beiden Begriffe deutlicher, als bisher geschehen, voneinander abzugrenzen: ausgehend von der Definition eines Objekts [Kapitel 3.2 Forderungen] besteht ein Lernobjekt aus wenigstens **zwei** Objekten und gibt eine Thematik wieder, die neben dem fachgebundenen Schwerpunkt zusätzlich Elemente einer technikspezifischen Klassifikation sowie lernzielrelevante Direktiven enthalten kann. Der Begriff Lernobjekt legt dabei nahe, dass es sich hierbei nicht um ein einzelnes Objekt handelt, sondern um mehrere Objekte, die thematisch zu einer geschlossenen Einheit gefügt sind und sich so gegenseitig ergänzen können. Wie schon erwähnt müssen hierbei sowohl für Objekte als auch für Lernobjekte detaillierte Beschreibungssätze vorliegen, die sich an beste-

[1] Die einzelnen Bausteine sind deshalb explizit genannt, weil in nachfolgenden Kapitel in exemplarischer Form darauf verwiesen wird.

henden Standards orientieren. Erst auf dieser Grundlage kann eine sinnvolle Verknüpfung von Objekten zu Lernobjekten erfolgen.

4.1.1 Standardisierung

Aspekte der Standardisierung sind gerade im Hinblick auf Interoperabilität, Portabilität und der damit verbundenen Strukturierung von grundsätzlicher Bedeutung. Das gilt nicht nur für die Beschreibung von Datensätzen, sondern für alle Ausführungen, die direkt oder indirekt damit verbunden sind. Bezogen auf die *datenbankbasierte Entwicklungsumgebung* sind das:

- Die Auszeichnungssprachen HTML und XHTML
- Die Beschreibungssprachen DC und RDF
- Die Integration gebräuchlicher MIME-Typen
- Die technikspezifische Klassifikation.

Bei der Nutzung existierender Standards muss jedoch berücksichtigt werden, dass nicht alle Standards, die durch entsprechende Institutionen (W3C, OASIS, IEEE, ANSI, etc.) verabschiedet wurden, auch von der Nutzergemeinschaft eingehalten bzw. in der Form und in dem Umfang, wofür sie geschaffen wurden, unterstützt werden. Die folgenden Ausführungen begründen den Einsatz standardisierter Elemente innerhalb der *datenbankbasierten Entwicklungsumgebung*.

Die Auszeichnungssprachen HTML und XHTML

Der Standard XML 1.0 stellt ein Beispiel für eine zwar durch die Nutzergemeinschaft akzeptierte, aber nur sehr wenig verbreitete Auszeichnungssprache dar [W3C, 2003c]. Obwohl sie vollständig beschrieben ist, stehen für eine Indexierung von XML-basierten Webseiten kaum Suchmaschinen zur Verfügung; als diffizil ist auch die Anbindung von XML-Datenbanken zu bewerten [Wilde, 2003]. Das gilt auch für die mit diesem Standard verbundene Syntax, die ausschließlich von neueren Versionen der Standard-Browser Netscape, Mozilla und dem Internet Explorer interpretiert wird. Noch problematischer wird die Situation, wenn Elemente, die bisher auch von älteren Browsern zuverlässig dargestellt wurden, wie beispielsweise CSS-Attribute in der Version CSS 1.0, jetzt innerhalb einer XML-DTD[1] definiert werden: es funktioniert nicht und erst der Einsatz von CSS 2.0 liefert zuverlässige Ergebnisse [Münz, 2002]. Einen möglichen Ausweg aus dieser Schwierigkeit bietet die Auszeichnungssprache XSL [W3C, 2003d]: sie verbindet CSS mit XML und hat die Hauptaufgabe, XML-Daten zu formatieren. Gegenwärtig besteht allerdings die Hauptanwendung von XSL in der Aufgabe, XML-Definitionen in HTML-Kodierungen zu transformieren[2]. Dieses Beispiel zeigt, wie schwierig der Einsatz neuer standardisierter Elemente ist und impliziert gleichzeitig einen immer wiederkehrenden Bezug zu HTML als elementare Auszeichnungssprache.

---

[1] Elemente einer XML-basierten Auszeichnungssprache werden unabhängig von den eigentlichen Daten durch externe DTDs definiert.

[2] HTML ist ein Dokument-Typ von SGML und existiert bereits seit den 70-er Jahren.

Daher lässt sich innerhalb der neuen und standardisierten Auszeichnungssprachen sowie deren Ergänzungen feststellen, dass HTML die am häufigsten eingesetzte Auszeichnungssprache[1] des Web ist. Auf der einen Seite ist HTML als Dokumentdefinition von SGML eine alte, gewachsene Sprache, die in gewisser Weise, nicht zuletzt wegen ihres statischen Charakters, als antiquiert bezeichnet werden kann; auf der anderen Seite ist HTML eine in ihrer Syntax überschaubare und leicht zu erlernende Sprache, die mittlerweile in der Version 4.01 vorliegt und sich offensichtlich bewährt hat. Es kann davon ausgegangen werden, dass auch künftig jeder Browser in der Lage sein wird, die meisten HTML-Elemente dieser Sprache in korrekter Form wiederzugeben. Da HTML-Konstrukte jedoch auch in unstrukturierter Form[2] vorliegen können, was sogar von den meisten Browsern durch entsprechende Fehlertoleranzen akzeptiert wird, hat das W3C daher Bestrebungen unternommen, HTML in Richtung einer deutlich strukturierteren Auszeichnungssprache zu überarbeiten. Das Ergebnis dieser Bemühungen heißt XHTML.

Diese Auszeichnungssprache schlägt eine Brücke zwischen HTML und XML, wobei im wesentlichen die für XML-Sprachen typische Syntax, wie z. B. ein strenges Einhalten des HTML-Grundgerüsts und die Einbindung von Abschluss-Tags eingefordert werden. Z. Zt. bereitet das W3C XHTML als sogenanntes 'working draft' in der Version 2.0 vor [W3C, 2003e]. Da XHTML sowohl zukunftsweisend ist, als auch an bewährte Standards in leicht verständlicher Form anknüpft, ist es **die** Auszeichnungssprache der *datenbankbasierten Entwicklungsumgebung* und wird daher nicht nur für die Generierung sondern auch für den Austausch webgebundener Elemente innerhalb der User-Community[3] und extern anzusprechenden Lern- und Kommunikationplattformen eingesetzt.

Die Beschreibungssprachen DC und RDF

Die Wahl von XHTML begünstigt den Einsatz von RDF als Beschreibungssprache für eine Implementierung strukturierter Datensätze. Da RDF selbst keine syntaktische sondern eine semantische Konzeption für Metadaten bietet, liegt es nahe, standardisierte Metadaten zu implementieren [Ferber, 2003]. Standardisierte Metadaten müssen neben den Möglichkeiten, die konventionelle Metatag-Elemente bieten, wie z. B. *Keywords*, *Description*, *Language*, *Publisher*, *Author*, etc. weitere Möglichkeiten einer differenzierten Beschreibung von Objekten eröffnen. So sollten z. B. Metatag-Elemente vorgesehen sein, die einzelne Objekte thematisch verwandten oder ähnlichen Objekten zuordnen können; ebenso sollten Element-Verfeinerungen möglich sein, die eine spezifische Beschreibung bestimmter Eigenschaften von Objekten erlauben. Schließlich sollte ein solcher Datensatz durch frei definierbare, neue Elemente erweiterbar und sowohl für eine Beschreibung von Objekten, als auch von Modulen gleichermaßen gut geeignet sein.

---

[1] Diese Aussage lässt sich beispielsweise durch die Hauptsuchmaschine Google mit der Eingabe von **allinurl: ".html"** überprüfen: Ergebnis ca. 492 Mio Seiten mit der Datei-Endung html. Dazu kommen noch einmal 318 Mio Seiten mit der Datei-Endung htm.

[2] Hiermit ist beispielsweise das Fehlen von Abschluss-Tags gemeint.

[3] Gesamtheit der registrierten Nutzer der *datenbankbasierten Entwicklungsumgebung*.

Mit Verweis auf vorhergehende Ausführungen [Kapitel 3.1.3 Strukturierte Datensätze und Kapitel 3.2.1, Ansatzpunkt] sollen im folgenden die Empfehlungen nach DCMI unterstützt werden. DC ist sowohl für eine Anbindung an (X)HTML- als auch an RDF-Elemente gleichermaßen gut geeignet, um standardisierte semantische Informationen über Inhalte zu liefern.

Ein Datensatz von **15 Basis-Elementen**, sog. 'Unqualified Elements', wie ihn DC bietet, reicht im wesentlichen für eine angemessene Beschreibung einzelner Objekte und Module aus und bildet die Grundlage für eine strukturierte Einordnung von Objekten, Lernmodulen, Lerneinheiten und Lehrgängen, wobei sowohl eine technikwissenschaftliche Klassifikation, als auch eine Berücksichtigung vorlesungs- und unterrichtsrelevanter Kriterien realisierbar ist.

Die Konventionen nach DC werden ausdrücklich vom W3C unterstützt. Tabelle 2 zeigt die 15 Basis-Elemente, die in Form von Meta-Daten in (X)HTML-Dokumente integriert werden bzw. über RDF-Dateien verfügbar sein können.

Nr.	Basis-Element[1]	Encoding-Schema	Beschreibung
1	DC.Title	Freier Text	Titel, Name des Dokuments oder der Ressource
2	*DC.Creator*	Freier Text	Autor, Verfasser, Urheber
3	*DC.Subject*	Freier Text oder Bezug zu LCSH	Gegenstand
4	DC.Description	Freier Text	Inhaltswiedergabe, Beschreibung
5	DC.Publisher	Freier Text	Verlag, Herausgeber, veröffentlichende Institution
6	DC.Contributor	Freier Text	Sonstige an der Erstellung und Veröffentlichung beteiligte Personen
7	*DC.Date*	W3C-DTF	Datum der Erstellung, der letzten Änderung oder der Begrenzung der Gültigkeit [W3C-DTF, 1997]
8	DC.Type	Freier Text und/oder DCMI Type	Zuordnung zu einer bestimmten Klasse, einem bestimmten Genre [DCMIType, 2003]
9	*DC.Format*	IMT	Dateiformat nach dem MIME-Standard
10	*DC.Identifier*	Freier Text oder URI	eindeutige Kennzeichnung, z. B. URI, ISBN, sonstige ID
11	DC.Source	Freier Text und/oder URI	bei nichtorginären Beiträgen die Quelle bzw. das Original
12	DC.Language	ISO 3166	Sprache, in der das Dokument vorliegt [IETF, 1995]
13	DC.Relation	Freier Text und/oder URI	Einordnung in andere Dokumente oder Zuordnung zu anderen Ressourcen, Hyperlinks
14	DC.Coverage	Freier Text	spezifische Zuordnung (organisatorische Abdeckung)
15	DC.Rights	Freier Text und/oder URI	Verweis auf die Nutzungsbedingungen entsprechend dem Urheberrecht

Tabelle 2: Die 15 Basis-Elemente mit Encoding-Schemata nach DC

Durch die DCMI werden für die oben dargestellte Übersicht verschiedene Schemata zur Encodierung der einzelnen Basis-Elemente empfohlen. Sie sind jedoch nicht verpflich-

[1] Die *kursiv* gedruckten Basis-Elemente (Tabelle 2) bzw. Element-Verfeinerungen (Tabelle 6) werden durch die *datenbankbasierte Entwicklungsumgebung* automatisch generiert. Alle anderen Eintragungen müssen vom jeweiligen Nutzer selbst durchgeführt werden.

tend und lassen sich durch eigene, speziellere Schemata ergänzen bzw. ersetzen [DCMI, 2003].

Integration unterschiedlicher MIME-Typs

Sowohl der DC- als auch der RDF-Standard ermöglichen eine Einbindung unterschiedlicher Medientypen in Form von standardisierten MIME-Typen [IETF, 1995; IETF, 1999]. Der MIME-Typ selbst wird dabei durch das Metatag **DC.Format** eingebunden. Er umfasst im wesentlichen Textdateien, Grafikdateien, Videodateien, Audiodateien und Applikationen[1].

Medientyp	MIME-Typ	Datei-Endung	Bedeutung
Text	`text/html`	`*.htm, *.html`	HTML-Dateien
Text	`text/plain`	`*.txt`	Text-Dateien
Graphik	`image/gif`	`*.gif`	GIF-Dateien
Graphik	`image/jpeg`	`*.jpeg, *.jpg`	JPEG-Dateien
Video	`video/mpeg`	`*.mpeg, *.mpg`	MPEG-Dateien
Video	`Video/x-msvideo`	`*.avi`	AVI-Dateien
Audio	`Audio/x-midi`	`*.mid, *.midi`	MIDI-Dateien
Audio	`audio/x-wav`	`*.wav`	WAV-Dateien
Application	`Applicatiom/pdf`	`*.pdf`	PDF-Dateien
Application	`application/octet-stream`	`*.class`	Java-Applets
Application	`application/x-shockwave-flash`	`*.swf`	Flash-Dateien
Application	`application/zip`	`*.zip`	ZIP-Dateien

Tabelle 3: Gebräuchliche MIME-Typen und deren Bedeutung

Tabelle 3 zeigt eine Übersicht der MIME-Typen, die im Internet am häufigsten verwendet werden. Innerhalb der *datenbankbasierten Entwicklungsumgebung* wurde jedoch bewusst auf einige Medientypen verzichtet: so ist die MIME-Type `application/pdf` für PDF-Dateien besser in digitalen Bibliotheken wie z. B. MILESS einsetzbar, da er sich nicht direkt lesbar in den Inhalt von (X)HTML-Dateien integrieren lässt. Weitere MIME-Typen, wie `video/mpeg`, `audio/x-wav` und `application/x-shockwave-flash` sind kritisch in der Anwendung, wenn keine Zusatzsoftware (Player, Plug-Ins, etc.) installiert ist. Andere Medientypen aus den Bereichen Video und Audio, wie `video/x-msvideo` für AVI-Dateien oder `audio/x-midi` für MIDI-Dateien werden erst bei deutlicher Nachfrage durch die User-Community implementiert, da sie (noch) nicht nachgefragt wurden.

[1] Applikationen sind Dateien, die an ein bestimmtes Programm gebunden sind, wie z. B. Java-Applets oder Flash-Animationen.

Technikspezifische Klassifikation

Schon am Anfang dieser Arbeit wurden Aspekte einer technikspezifischen Ergänzung angesprochen [Kapitel 2 Ausgangslage]. Dabei ging es hauptsächlich um eine Klassifikation technischer Sachsysteme auf der Basis systemtheoretischer Betrachtungen. Die folgende Tabelle 4 zeigt die Klassifikation technischer Systeme und liefert eine Klassifikations-Matrix mit 9 Feldern, die jeweils eine bestimmte Klasse von Sachsystemen (Verfahren) bestimmen [Ropohl, 1999]. Einzelne Sachsysteme lassen sich wiederum in standardisierter Form unterteilen. So existiert beispielsweise in der Fertigungstechnik eine standardisierte Einteilung der Fertigungsverfahren nach der DIN 8580: *Urformen*, *Umformen*, *Trennen*, *Fügen*, *Beschichten* und *Stoffeigenschaften ändern*, während auf dem Gebiet der Verfahrenstechnik z. B. die DIN 28004 die Ausführung von RI-Fliessbildern verfahrenstechnischer Anlagen standardisiert [Graßmuck, 1994].

Funktionen ► *Attribute* ▼	**Wandlung (W)** (Produktionstechnik)	**Transport (T)** (Transporttechnik)	**Speicherung (Sp)** (Speicherungstechnik)
Stoff (S) (Materialtechnik)	Verfahrenstechnik Fertigungstechnik	Fördertechnik Verkehrstechnik Tiefbautechnik	Behältertechnik Lagertechnik Hochbautechnik
Energie (E) (Energietechnik)	Energiewandlungstechnik	Energieübertragungstechnik	Energiespeicherungstechnik
Information (I) (Informationstechnik)	Informationsverarbeitungstechnik Mess-, Steuer- und Regelungstechnik	Informationsübertragungstechnik	Informationsspeicherungstechnik

Tabelle 4: Klassifikation technischer Sachsysteme

Neben dieser Klassifikation, die auf einer Standardisierung verschiedener technischer und ingenieurwissenschaftlicher Disziplinen beruht, gibt die Attribute-Funktionen-Matrix neben den vorgenannten Sachsystemen sog. Anwendungsfelder wieder. So sind z. B. im Anwendungsfeld *Information und Kommunikation* sämtliche Funktionen (W, T, Sp) für das Attribut Informationstechnik (I) zusammengefasst. Das Anwendungsfeld *Automation* ist universell und umfasst neben allen drei Funktionen (W, T, Sp) auch alle drei Attribute (S, E, I). Es ist in der folgenden Tabelle 5 aus Gründen der Übersichtlichkeit nicht eingezeichnet.

Funktionen ► / *Attribute* ▼	**Wandlung (W)** (Produktionstechnik)	**Transport (T)** (Transporttechnik)	**Speicherung (Sp)** (Speicherungstechnik)
Stoff (S) (Material-technik)	*Versorgung und Entsorgung*	*Transport und Verkehr*	
Energie (E) (E-nergietechnik)			
Information (I) (Informationstechnik)	*Information und Kommunikation*		

Tabelle 5: Attribute-Funktionen-Matrix (A-F-Matrix)

Bezogen auf die *datenbankbasierte Entwicklungsumgebung* bildet die A-F-Matrix das letzte standardisierte Element, welches für ein strukturiertes Ablegen von Objekten und Modulen notwendig ist und schließt somit den Aufbau eines standardisierten Fundaments für eine darauf aufbauende Konzeption ab.

4.1.2 Konzeption

Die im vorhergehenden Kapitel angesprochenen Elemente der Standardisierung sollen im folgenden in der Konzeption des modularen Ansatzes, bezogen auf die *datenbankbasierte Entwicklungsumgebung*, detailliert dargestellt werden. Ausgehend von **DC-Verfeinerungen** wird systematisch über den Einsatz von **HTML**, **XHTML** und **RDF** in exemplarischer Form zu der angestrebten **modularen Kombinierbarkeit** bei der Generierung von Lernmodulen, Lerneinheiten und Lehrgängen hingeleitet.

DC-Verfeinerungen

Jedes der in Tabelle 2 dargestellten Basis-Elemente bietet Element-Verfeinerungen, die als Dublin Core Qualifiers (DCQ) bezeichnet werden. Tabelle 6 zeigt diejenigen Verfeinerungen, die im Zusammenhang mit der *datenbankbasierten Entwicklungsumgebung* sinnvoll einsetzbar sind und durch die DCMI empfohlen und beschrieben werden.

Nr.	**Element-Verfeinerung**	**Encoding-Schema**	**Beschreibung**
2.1	*DC.Creator.Email*	RFC 2731	E-Mail-Adresse des Autors, Verfassers, Urhebers [IETF, 1999]
3.1	`DC.Subject.Keywords`	Freier Text	Inhaltlich relevante Schlüsselwörter
13.1	`DC.Relation.IsPartOf`	Freier Text	Ergänzende und thematisch verwandte Inhalte oder Bezüge
13.2	`DC.Relation.References`	Freier Text	Bezug zu speziellen Hinweisen oder Elementen

Tabelle 6: Element-Verfeinerungen nach DC

In Tabelle 7 und Tabelle 8 sind diejenigen Element-Verfeinerungen wiedergegeben, die speziell die *datenbankbasierte Entwicklungsumgebung* betreffen, und zwar aus den Bereichen **technikspezifischer** und **fachdidaktischer** Beschreibungen. Sie beziehen sich ausnahmslos auf die Element-Verfeinerung 13.2 aus Tabelle 6. Hier wurde von der Möglichkeit der DCMI Gebrauch gemacht, selbst entwickelte Element-Verfeinerungen zu verwenden, um spezielle Informationen zu integrieren [Ferber, 2003a]. Dies gilt insbesondere im Zusammenhang mit RDF und XML und ist für eine spätere systematische Einordnung von Modulen höherer Ordnung in Lern- und Kommunikationsplattformen, wie z. B. ILIAS, wichtig.

Nr.	**Spezielle Element-Verfeinerung**	**Encoding-Schema**	**Beschreibung**
13.2.1	`DC.Relation.References.Attributes`	Freier Text	Attribute (S, E, I)
13.2.2	`DC.Relation.References.Functions`	Freier Text	Funktionen (W, T, Sp)
13.2.3	`DC.Relation.References.Fields`	Freier Text	Anwendungsfelder
13.2.4	`DC.Relation.References.Processes`	Freier Text	Verfahren

Tabelle 7: Element-Verfeinerungen für technikspezifische Beschreibungen

Nr.	**Spezielle Element-Verfeinerung**	**Encoding-Schema**	**Beschreibung**
13.2.5	`DC.Relation.References.Theme`	Freier Text	Lernzielorientierung (Inhalt)
13.2.6	`DC.Relation.References.Cognition`	Freier Text	Komplexität
13.2.7	`DC.Relation.References.Affection`	Freier Text	Internalisation
13.2.8	`DC.Relation.References.Psychomot`	Freier Text	Koordination
13.2.9	`DC.Relation.References.Objectives`	Freier Text	Schwerpunktsetzungen

Tabelle 8: Element-Verfeinerungen für fachdidaktische Beschreibungen

Die nachfolgende Abbildung dient einer besseren Übersicht und soll den durch Tabelle 4, Tabelle 7 und Tabelle 8 wiedergegebenen Zusammenhang visualisieren. In ihr sind die Abhängigkeiten von Attributen, Funktionen und Lernzielen dargestellt.

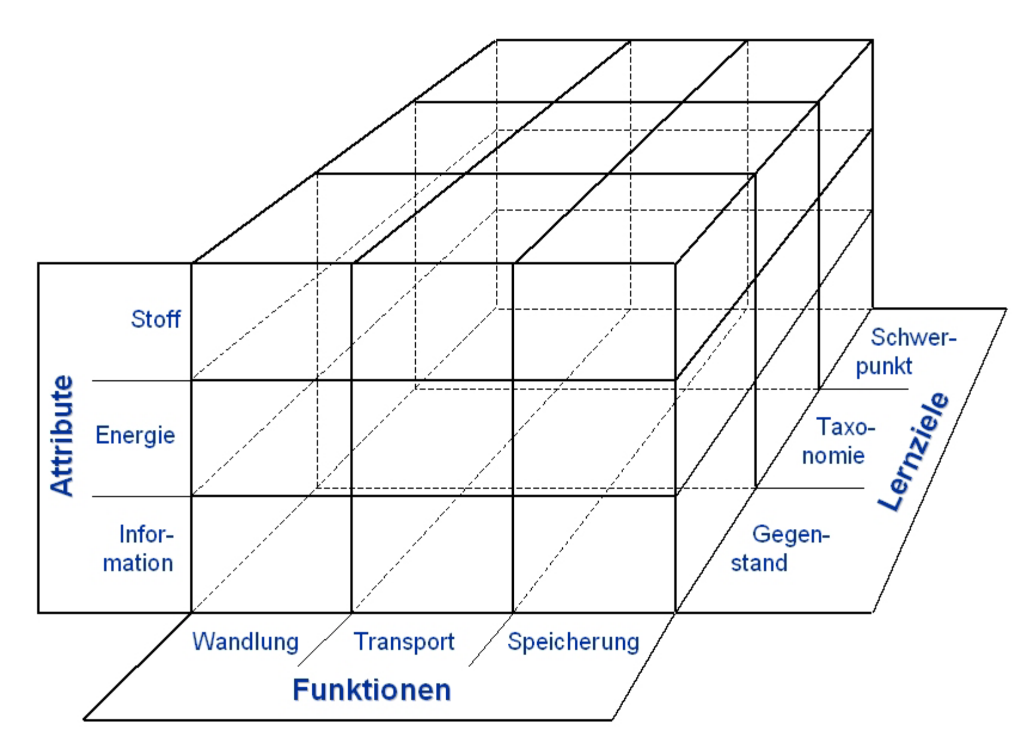


Abbildung 4: Abhängigkeiten: Attribute, Funktionen und Lernziele

Interne Formate HTML und XHTML

Eine Einbindung der Metatags nach DC erfolgt über die Auszeichnungssprache (X)HTML. Abbildung 5 gibt den typischen Aufbau der Metatag-Elemente wieder, die im Header, also zwischen den Tags `<head>` und `</head>` einer XHTML-Seite plaziert werden. Das Listing in Abbildung 5 **"Der Klappentelegraph von Murray"**[1] zeigt die Metatag-Implementierung nach DC, wie sie als parserfähiges[2] Eingabe-Format im Rahmen eines Eintrags als (X)HTML-Datei[3] von der *datenbankbasierten Entwicklungs-umgebung* erwartet wird.

---

[1] Den Bezug bildet hierbei die HTML-Datei **klappentelegraph_beschreibung.htm** .

[2] Als parsing wird hier ein zum Zwecke des Einlesens grammtisches Zerlegen einzelner Zeilen bezeichnet.

[3] Hier wird von (X)HTML gesprochen, da die zwischen den `<head>` Tags plazierten Elemente sowohl in HTML als auch in XHTML identisch sind.

```
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC
  "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta name="DC.Title" content="Der Klappentelegraph von Murray, Beschrei-
bung">
<meta name="DC.Creator" content="Jürgen Wehling">
<meta name="DC.Creator.Email" content="juergen.wehling@uni-essen.de">
<meta name="DC.Subject" content="Klappentelegraph">
<meta name="DC.Subject.Keywords" content="Klappentelegraph, Murray, opti-
scher Telegraph, optische Nachrichtentechnik, historische Aspekte">
<meta name="DC.Description" content="Der Klappentelegraph von Murray war
ein optischer Telegraph. Er wurde von Lord George Murray konstruiert und
1795 zwischen London und Deal (an der schmalsten Stelle des Ärmelkanals)
installiert. Abgelesen wurde dieses frühe Instrumentarium der optischen
Nachrichtentechnik mit Hilfe eines Fernrohres.">
<meta name="DC.Publisher" content="http://www.mmdb-tu.de">
<meta name="DC.Contributor" content="Robert Asano, Denis Bhattacharya,
Wolfgang Haupt">
<meta name="DC.Date" content="2001-04-30">
<meta name="DC.Type" content="Kurs/Onlinekurs/Virtuelles Seminar">
<meta name="DC.Format" content="text/html">
<meta name="DC.Identifier" content="http://fred2.technik-didaktik.uni-
essen.de/~mmdb-tu/wehling/objekte/klappentelegraph_beschreibung.htm">
<meta name="DC.Source" content="V. Aschoff - Geschichte der Nachrichten-
technik, Berlin: Springer 1984; G.J. Holzmann, B. Pehrson - Optische Te-
legraphen und die ersten Informationsnetze, in: Spektrum der Wissen-
schaft, März 1994; R. Oberliesen - Information, Daten und Signale; Bib-
liothek der britischen Admiralität, London">
<meta name="DC.Language" content="de">
<meta name="DC.Relation" content=" ">
<meta name="DC.Relation.IsPartOf" content="klappentelegraphdt.swf, mur-
ray2.class, murraydiagr.jpg">
<meta name="DC.Coverage" content="Informationsumsatz">
<meta name="DC.Rights" content="http://www.tud.uni-essen.de">
</head>
<body> . . . <body>
</html>
```

Abbildung 5: Implementierung konventioneller Metatags nach DC

Innerhalb der *datenbankbasierten Entwicklungsumgebung* wird der Nutzer ausschließlich mit den Auszeichnungssprachen HTML 4.0 bzw. XHTML 1.0 konfrontiert. Sämtliche in Tabelle 3 wiedergegebenen Medientypen lassen sich, von den angesprochenen Ausnahmen abgesehen, mit der durch Abbildung 5 beispielhaft dargestellten Syntax in strukturierter Form beschreiben.[1] Der Beschreibungssatz nach DC **kann** bei (X)HTML-Dateien in den Header integriert werden.

Es bietet sich jedoch auch die Möglichkeit, diesen Beschreibungssatz als **externe Datei** auszuführen, was gerade bei den übrigen Medientypen (Graphik, Video, Audio, etc.)

[1] DC.Relation ist in diesem Fall ein leeres Feld. Ein übergeordneter Bezug wird erst bei der Generierung von Lehrgängen hergestellt.

nicht nur von Vorteil, sondern gänzlich unvermeidbar ist. So lässt sich beispielsweise der Medientyp 'Application' (in Abbildung 6 als Applet *murray2.class*[1] dargestellt) durch eine externe RDF-Datei (*murray2.rdf*) in einer einfachen für die *datenbankbasierte Entwicklungsumgebung* geeigneten Form beschreiben.

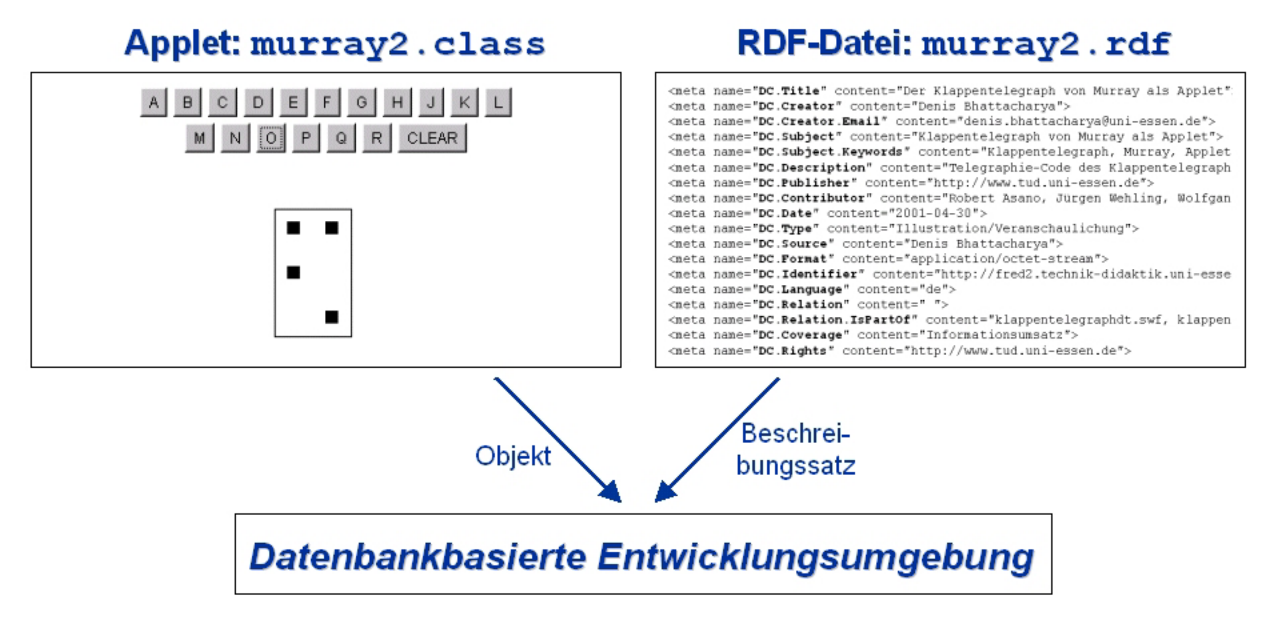


Abbildung 6: Medientyp *murray2.class* mit externer Datei *murray2.rdf*

Externe RDF-Referenz

Eine solche Objektreferenz sollte sich am RDF-Standard orientieren. Während es für eine externe Verarbeitung (Export) unumgänglich ist, so ist es für die interne Verarbeitung durch die *datenbankbasierte Entwicklungsumgebung* jedoch nicht notwendig. Hier kommt ein nutzerfreundliches und leicht verständliches Format zum Einsatz, das durch Abbildung 5 für (X)HTML-Dateien bzw. durch Abbildung 6 für Applikationen wiedergegeben ist.

Neben einem konsequenten Einsatz von DC-basierten Metatags werden alle durch die *datenbankbasierte Entwicklungsumgebung* erzeugten Lernobjekte in der Auszeichnungssprache XHTML exportiert. Die Einbindung von Metatags in XHTML erfolgt durch RDF-Implementierungen, die für den Export durch die *datenbankbasierte Entwicklungsumgebung* auf der Basis vorhandener Datenbeschreibungssätze automatisch generiert werden [DCMI, 2002b]. Abbildung 7 zeigt das Beispiel aus Abbildung 5 mit RDF-Elementen, die für einen Export aufbereitet sind.

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
```

[1] Das Applet `murray2.class` wurde auf der Basis von Codiertabellen der Britischen Admiralität von Herrn D. Bhattcharya entwickelt.

```
          xmlns:dc="http://purl.org/dc/elements/1.1/"
          xmlns:dcq="http://purl.org/dc/qualifiers/1.1/">
   <rdf:Description rdf:about=" http://fred2.technik-didaktik.uni-
   essen.de/~mmdb-tu/bhattacharya/objekte/murray.htm">
     <dc:title>Der Klappentelegraph von Murray, Beschreibung</dc:title>
     <dc:creator>
       <rdf:Description>
         <rdf:value>Jürgen Wehling</rdf:value>
         <dcq:email>juergen.wehling@uni-essen.de</dcq:email>
       </rdf:Description>
     </dc:creator>
     <dc:subject
       <rdf:Description>
         <rdf:value>Klappentelegraph</rdf:value>
         <dcq:keywords>Klappentelegraph, Murray, optischer Telegraph, op-
         tische Nachrichtentechnik, historische Aspek-
         te</dc:subject.keywords</dcq:keywords>
       </rdf:Description>
     </dc:subject>
     <dc:description>Der Klappentelegraph von Murray war ein optischer Te-
     legraph. Er wurde von Lord George Murray konstruiert und 1795 zwi-
     schen London und Deal (an  der schmalsten Stelle des Ärmelkanals) in-
     stalliert. Abgelesen wurde dieses frühe Instrumentarium der optischen
     Nachrichtentechnik mit Hilfe eines Fernrohres.</dc:description>
     <dc:publisher>http://www.mmdb-tu.de</dc:publisher>
     <dc:contributor>Robert Asano,Denis Bhattacharya, Wolfgang
     Haupt</dc:contributor>
     <dc:date>2001-04-30</dc:date>
     <dc:type>Kurs/Onlinekurs/Virtuelles Seminar</dc:type>
     <dc:format>text/html</dc:format>
     <dc:identifier>http://fred2.technik-didaktik.uni-essen.de/~mmdb-
     tu/wehling/objekte/klappentelegraph_beschreibung.htm
     </dc:identifier>
     <dc:source>V. Aschoff - Geschichte der Nachrichtentechnik, Berlin:
     Springer 1984; G.J. Holzmann, B. Pehrson - Optische Telegraphen und
     die ersten Informationsnetze, in: Spektrum der Wissenschaft, März
     1994; R. Oberliesen - Information, Daten und Signale; Bibliothek der
     britischen Admiralität, London</dc:source>
     <dc:language>de</dc:language>
     <dc:relation>
       <rdf:Description>
         <rdf:value> </rdf:value>
         <dcq:ispartof>klappentelegraphdt.swf, murray2.class, murray-
         diagr.jpg</dcq:ispartof>
       </rdf:Description>
     </dc:relation>
     <dc:coverage>Informationsumsatz</dc:coverage>
     <dc:rights>http://www.tud.uni-essen.de</dc:rights>
   </rdf:Description>
</rdf:RDF>
```

Abbildung 7: Implementierung RDF-bezogener Metatags nach DC

Durch die Abbildung 6 und die Abbildung 7 ist verdeutlicht worden, dass es sich bei einer Implementierung von Metatags auch um externe Angaben handeln kann, die in Form einer Beschreibungsdatei vorliegen. Im Falle einer XHTML-Datei sollte die modifizierte Header-Zeile

```
<head profile="klappentelegraph_beschreibung.rdf">
```

auf eine extern vorliegende Beschreibungsdatei mit RDF-Elementen verweisen, d. h. der komplette Header wird ausgelagert. Insbesondere bei der Eintragung von Objekten

erscheint diese Vorgehensweise sinnvoll: zu jedem Objekt existiert eine externe Datei mit dem zugehörigen Beschreibungssatz. So gehört z. B. zum Bild der "Darstellung des historischen Klappentelegraphen von Murray" (Datei: *klappentelegraph_murray.jpg*) eine namensgleiche Beschreibungsdatei mit der Dateiendung RDF:

```
klappentelegraph_murray.jpg
klappentelegraph_murray.rdf
```

Der Vorteil dieser Vorgehensweise wird bei der Betrachtung einer Vielzahl ähnlicher oder themenverwandter Objekte deutlich:

Es müssen nur noch wenige Zeilen des Beschreibungssatzes editiert werden.

Es ist jedoch zu berücksichtigen, dass es in den Spezifikationen zu HTML 4.0 und XHTML 1.0 weder Angaben darüber gibt, **wie** der Aufbau einer externen Beschreibungsdatei auszusehen hat, noch **welche** Dateiendung zu verwenden ist [Münz, 2002a]. Ganz allgemein wird im Zusammenhang mit der *datenbankbasierten Entwicklungsumgebung* immer die Dateiendung RDF verwendet. Bei der Eintragung von Objekten wird der Nutzer jedoch nicht mit der komplexen RDF-Syntax von reinem XHTML konfrontiert. Das ist in diesem Fall nicht erforderlich, da sog. Template-Generatoren durch die *datenbankbasierten Entwicklungsumgebung* zur Verfügung gestellt werden, die den Nutzer gezielt anleiten.

Exportformat XHTML

Wenn es jedoch um den Export einer Lerneinheit oder eines Lehrgangs[1] geht, ist das Format auf XHTML mit RDF-Elementen festgelegt. Die zu exportierende XHTML-Datei (*klappentelegraph_beschreibung.htm*) mit Verweis auf die in Abbildung 7 dargestellte zugehörige RDF-Datei ist in Abbildung 8 wiedergegeben.

```
<?xml version="1.0"?>
<!DOCTYPE html PUBLIC
  "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head profile="klappentelegraph_beschreibung.rdf">
<title>Der Klappentelegraph von Murray, Beschreibung</title>
</head>
<body>
 . . .
<body>
</html>
```

Abbildung 8: Für den Export generierte XHTML-Datei

Einsatz konventioneller Metatags

[1] Es ist nicht vorgesehen, Lernmodule für den Export aufzubereiten. Sie bieten nicht registrierten Nutzern lediglich einen Anreiz, sich registrieren zu lassen: Schließlich steht registrierten Nutzern die wesentlich komfortablere Möglichkeit einer Generierung von Lerneinheiten zur Verfügung.

Für eine Veröffentlichung über das Internet ist es jedoch sinnvoller, an Stelle von DC-basierten Metatags, die für den Export zu Lern- und Kommunikationsplattformen wichtig sind, den Einsatz konventioneller Metatags vorzuziehen. Diesem Umstand wird dadurch Rechnung getragen, dass auch die Möglichkeit besteht, Lernobjekte[1] in dieser Form zur Verfügung zu stellen. Die *datenbankbasierte Entwicklungsumgebung* bietet zu diesem Zweck eine Reihe von Hilfsmitteln zur Konvertierung an. Hierbei ist jedoch zu beachten, dass längst nicht alle Elemente, die durch DC beschrieben wurden, auch in adäquater Form umgesetzt werden können, wie die Abbildung 9 zeigt. Welche Elemente indexierbar sind, ist nicht standardisiert.

```
<meta name="author" content="Jürgen Wehling">
<link rev=made href="mailto:juergen.wehling@uni-essen.de">
<meta name="keywords" content="Klappentelegraph, Murray, optischer Tele-
graph, optische Nachrichtentechnik, historische Aspekte">
<meta name="description" content="Der Klappentelegraph von Murray war ein
optischer Telegraph. Er wurde von Lord George Murray konstruiert und 1795
zwischen London und Deal (an  der schmalsten Stelle des Ärmelkanals) in-
stalliert. Abgelesen wurde dieses frühe Instrumentarium der optischen
Nachrichtentechnik mit Hilfe eines Fernrohres.">
<meta name="publisher" content="http://www.tud.uni-essen.de">
<meta name="owner" content="http://www.tud.uni-essen.de">
<meta name="lastupdated" content="2001-04-30">
<meta http-equiv="content-type" content="text/html; charset=iso-8859-1">
<meta name="identifier-url" content="http://fred2.technik-didaktik.uni-
essen.de/~mmdb-tu/wehling/objekte/klappentelegraph_beschreibung.htm">
<meta name="content-language" content="de">
<meta name="copyright" content="http://www.tud.uni-essen.de">
```

Abbildung 9: Gebräuchliche konventionelle Metatag-Angaben

Auch hier werden die Metatags wieder im Header zwischen **<head>** und **</head>** platziert bzw. durch den Verweis auf eine externe Datei referenziert. Es sei darauf hingewiesen, dass in Abbildung 9 nur die gebräuchlichsten konventionellen Metatags aufgeführt sind und diese zusätzlich abhängig von den verwendeten Suchmaschinen sind, über welche die HTML-Seite veröffentlicht werden soll; jedoch längst nicht alle Suchmaschinen indexieren HTML-Seiten auch nach den dort aufgeführten Metatags!

[1] Lerneinheiten und Lehrgänge sind durch konventionelle Metatags nicht mehr ausreichend beschreibbar.

Modulare Kombinierbarkeit

An dieser Stelle soll die Konzeption, welche der modularen Kombinierbarkeit zugrunde liegt, durch Schaubilder erläutert werden[1]. Während bei der Generierung eines Lernmoduls das Ergebnis einer Datenbankabfrage in ungeordneter Form vorliegt, ist bei der Generierung einer Lerneinheit oder eines Lehrgangs dieses Ergebnis geordnet und durch themenverwandte Empfehlungen erweitert. Aus diesen und aus entwicklungstechnischen Gründen wird ein Lernmodul auch als *Sammlung*, eine Lerneinheit als *Proposal* und ein Lehrgang, der auf der Basis von Lerneinheiten entstanden ist, als *Variante* bezeichnet [Kapitel 6.2. Begriffsdefinitionen und -analogien]

Die grundlegende Konzeption bei der Generierung von Lernobjekten folgt immer dem gleichen Prinzip: durch die nutzerspezifische Angabe ausgewählter Objekt-IDs wird eine Objekte-Liste angelegt, die im Falle der Generierung einer Lerneinheit oder eines Lehrgangs jederzeit individuell bearbeitet werden kann. Auf der Grundlage der Reihenfolge der in dieser Objekte-Liste aufgenommenen Objekt-IDs wird schließlich das Lernobjekt selbst generiert[2].

Bei der Eintragung eines so generierten Lernobjekts in die Datenbank müssen zusätzlich spezielle, lernobjektabhängige Kriterien beachtet werden, die in den Hinweisen jeweils am Ende der nachfolgenden Abbildungen formuliert sind.

1 Eine wesentliche Funktion der *datenbankbasierten Entwicklungsumgebung* ist, wie schon mehrfach erwähnt, die Generierung von Lernobjekten: damit ist die modulare Kombinierbarkeit einfacher Objekte/Module zu Lernmodulen, Lerneinheiten und Lehrgängen gemeint.

2 Bei der Generierung von Lehrgängen bildet nicht eine Liste von Objekten, sondern eine Liste von Lerneinheiten die Grundlage.

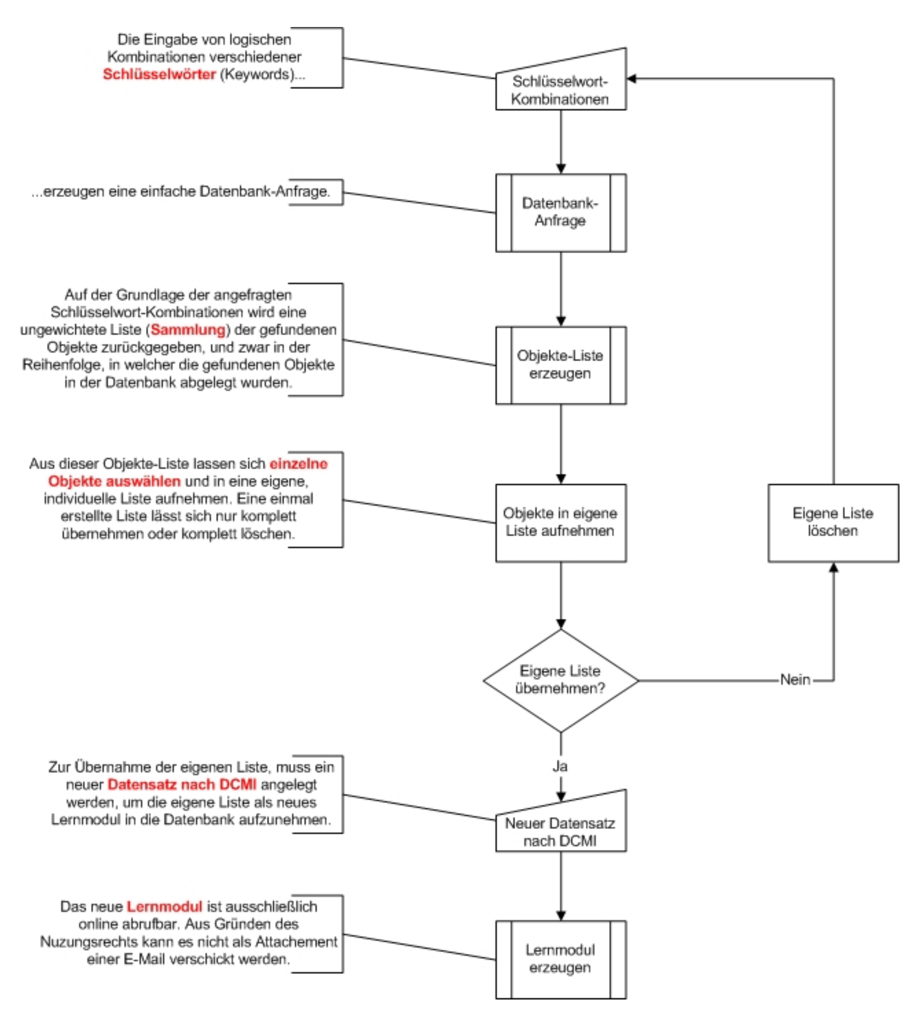


Abbildung 10: Konzeption zur Generierung eines Lernmoduls

Hinweis: Die Kombination von zwei oder mehr **Objekten** erfordert bei einer Eintragung als neues **Lernmodul** auch einen neuen Datenbeschreibungssatz nach DCMI [Kapitel 3.2.1 Ansatzpunkt].

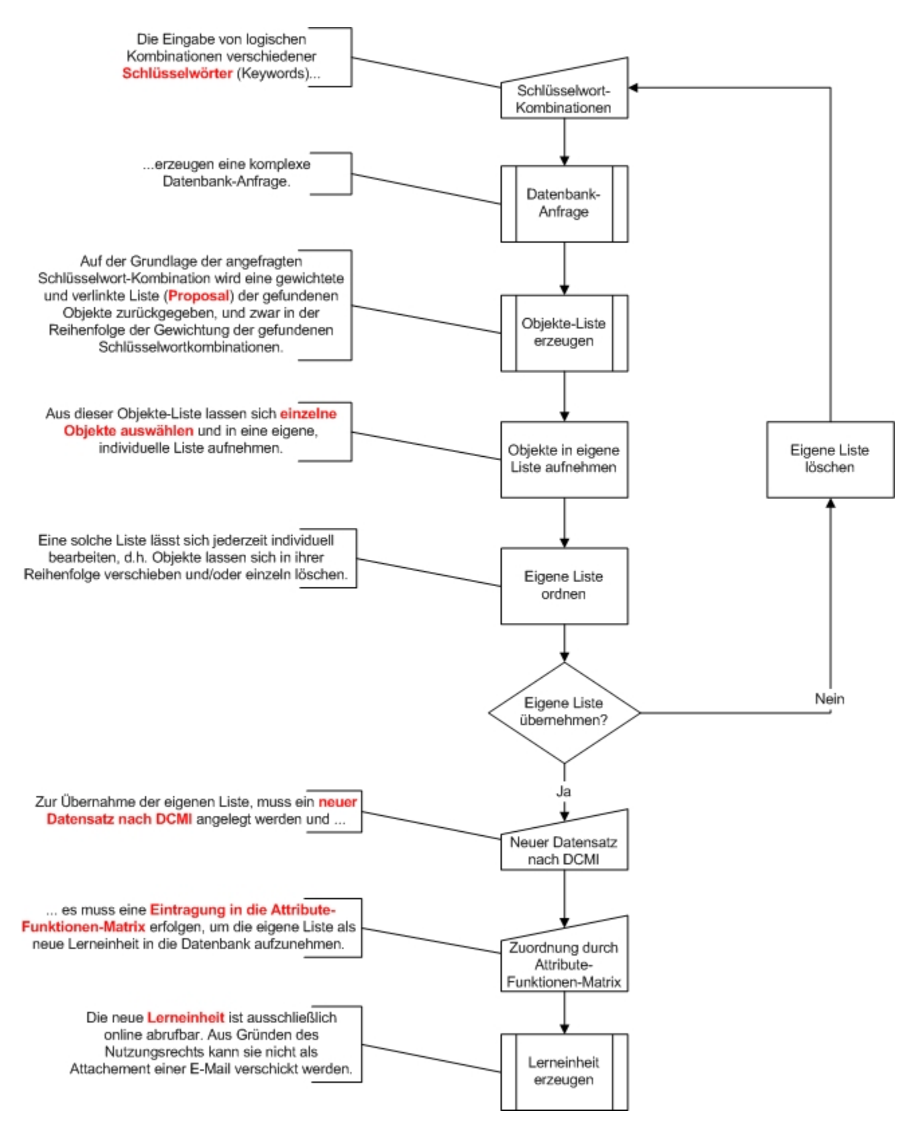


Abbildung 11: Konzeption zur Generierung einer Lerneinheit

Hinweis: Die Kombination von zwei oder mehr **themenverwandten Objekten** erfordert bei einer Eintragung als **Lerneinheit** einen neuen Datenbeschreibungssatz nach DCMI sowie eine technikspezifische Beschreibung (Attribute-Funktionen-Matrix) [Kapitel 3.2.1 Ansatzpunkt].

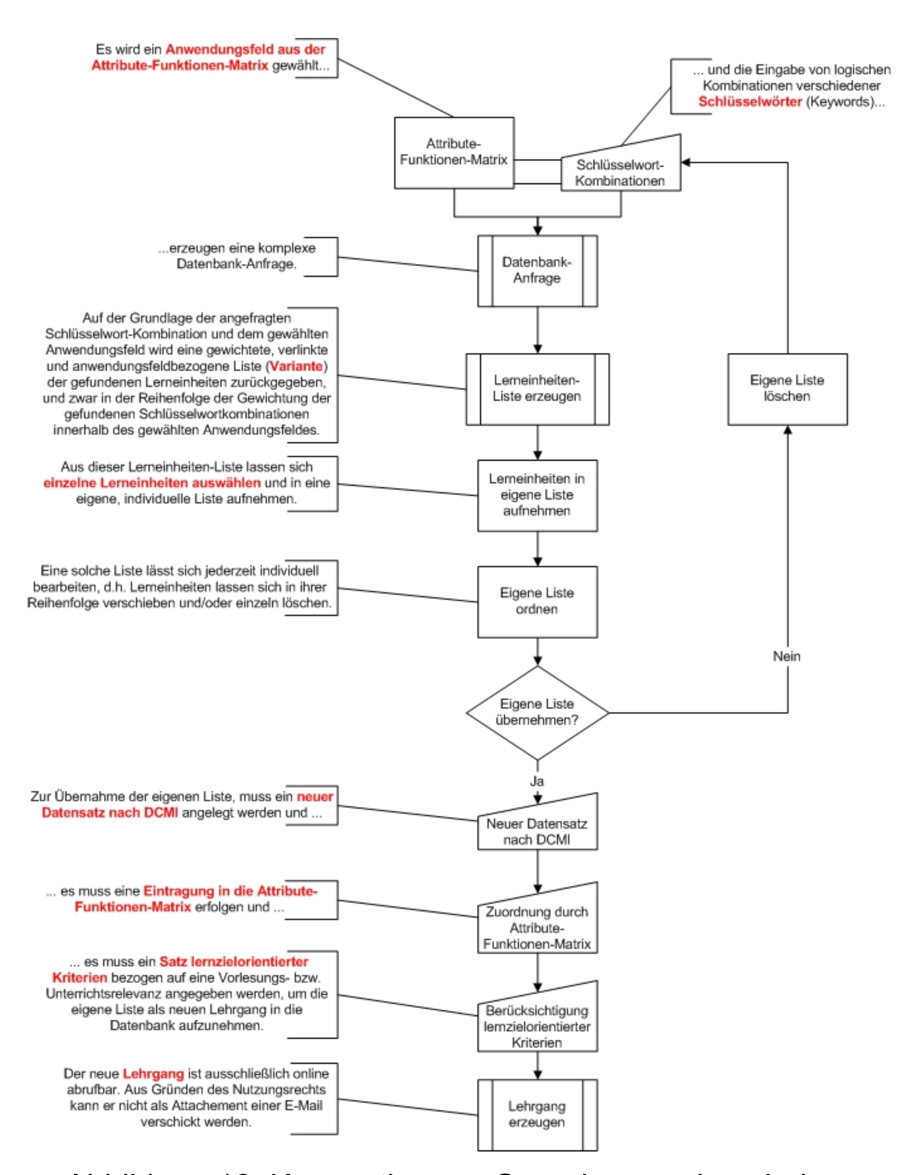


Abbildung 12: Konzeption zur Generierung eines Lehrgangs

Hinweis: Die Kombination von zwei oder mehr **themenverwandten Lerneinheiten** erfordert bei einer Eintragung als **Lehrgang** einen neuen Datenbeschreibungssatz nach DCMI und neben einer technikspezifischen (Attribute-Funktionen-Matrix) zusätzlich eine fachdidaktische Beschreibung (lernzielorientierte Kriterien) [Kapitel 3.2.1 Ansatzpunkt].

Die modulare Kombinierbarkeit von Lernobjekten ist ausschließlich **online** durchführbar.

Diese Netzbindung findet ihre Begründung zu einem Teil im Open-Source-Ansatz und den damit verbundenen **Programmiersprachen**, zum anderen Teil in den mit der *datenbankbasierten Entwicklungsumgebung* verbundenen **rechtlichen Problematiken**; außerdem soll das System jedem registriertem Nutzer orts- und zeitunabhängig zugänglich sein.

Die **Programmiersprachen** Perl und PHP werden im wesentlichen serverseitig interpretiert und ausgeführt, wodurch eine Offline-Generierung von Lernobjekten im Rahmen der *datenbankbasierten Entwicklungsumgebung* definitiv ausgeschlossen ist.

Bezüglich der **rechtlichen Problematiken** [Kapitel 3.1.1 Content] verpflichtet sich jedes Mitglied der User-Community bei seiner Registrierung, die Nutzung seiner Objekte/Module jedem anderen Mitglied innerhalb der *datenbankbasierten Entwicklungsumgebung* zu erlauben: einer Online-Nutzung zum Zweck der Lehre steht damit nichts mehr im Wege. Hierbei bildet eine Abtretungserklärung des UVM die Grundlage [UVM, 2003b]. Sollen Module jedoch exportiert werden, z. B. in Lern- und Kommunikationsplattformen, so ist zu beachten, dass weder das Modul noch die zugehörigen Objekte kommerziell genutzt werden dürfen [MSWF, 2003].

Eine globale Freigabe der *datenbankbasierten Entwicklungsumgebung* soll jedem an Technik Interessiertem einen leichten Zugang zu technischen Inhalten erlauben. Eine solche Freigabe ist durch die bilingual existierenden graphischen Frontends realisierbar und auch vorgesehen. So besteht die Möglichkeit, neben dem fachwissenschaftlichen Inhalt auch den damit verbundenen fachdidaktischen Gehalt zu fördern. Auf diese Weise kann zusätzlich eine Qualitätssicherung der Objekte/Module erfolgen.

4.1.3 Integration

Um den modularen Ansatz in seiner vollständigen Integration von Datenbank, graphisch orientierten Nutzerschnittstellen und implementierten Tools in geeigneter Form darzustellen, sind im folgenden zum besseren Verständnis an wichtigen Textstellen Hyperlinks zur referentiellen Ergänzung eingefügt. Als ein Baustein des L.A.M.P.-Ansatzes wurde das RDBMS MySQL gewählt. Auf dieser Grundlage wurde die eigentliche Datenbank der MMDB-TU, die *mmdb*, angelegt[1]. Der dabei zugrunde liegende modulare Ansatz spiegelt sich nicht nur in den durch die Datenbank *mmdb* zur Verfügung gestellten Objekten/Modulen wieder, sondern er findet sich auch in wesentlichen Elementen der gesamten Konzeption.

[1] Mit MMDB-TU wird die graphisch orientierte Nutzerschnittstelle für aktive Datenbank-Zugriffe bezeichnet. Die dahinter stehende Datenbank selbst heisst *mmdb*.

Entitäten

Als relationale Datenbank ist die *mmdb* streng modular aufgebaut, wobei ihre Modularität durch Entitätsmengen im zugehörigen ER-Diagramm gekennzeichnet ist [Kapitel 4.2.3 Nutzerschnittstelle MMDB-TU]. Hierbei ist jedoch nicht gemeint, dass sich Entitätsmengen in irgendeiner Form neu kombinieren oder austauschen lassen: durch Maßnahmen der Normalisierung wird die Möglichkeit geboten, weitere Entitätsmengen modular in einfacher Form zu ergänzen, wenn sie notwendig oder durch die User-Community gefordert werden sollten. So lassen sich beispielsweise Entitätsmengen für exportierte Lerneinheiten oder Lehrgänge sehr schnell in die Datenbank *mmdb* aufnehmen und über entsprechende Primär- und Fremdschlüssel[1] integrieren.

Nutzerschnittstellen

Der Nutzer selbst tritt bei seiner Arbeit nur indirekt mit der Datenbank *mmdb* in Kontakt. Dem modularen Ansatz folgend stellt die *datenbankbasierte Entwicklungsumgebung* ihm verschiedene Schnittstellen für eine Arbeit mit der Datenbank *mmdb* zur Verfügung. Diese Nutzerschnittstellen sind als graphische Benutzeroberflächen durch die Skriptsprachen Perl und PHP[2] realisiert und beziehen sich auf elementare Prozesse[3] der *datenbankbasierten Entwicklungsumgebung*, wobei Perl für **aktive** und PHP für **passive Datenbank-Zugriffe** eingesetzt wird. Ein aktiver Datenbank-Zugriff ist hierbei ein schreibender (*add*) oder programmausführender[4] (*execute*), ein passiver ein lesender (*access*) Prozess. Die modulare Konzeption dieser Datenbank-Zugriffe sieht folgendermaßen aus:

Aktive Datenbank-Zugriffe:

- Eintragung und Korrektur von Nutzerdaten
- Eintragung, Korrektur und Löschen von Objekten
- Eintragung, Korrektur und Löschen von Lernmodulen, Lerneinheiten und Lehrgängen
- Parsing und Eintragung von kompletten Objekt-Archiven[5]

Im wesentlichen beschränken sich aktive Datenbank-Zugriffe auf die Prozesse *Eintragung und Korrektur von Objekten/Modulen*. Die damit verbundenen durch Perl-Programme realisierten Frontends sind in ihrem graphischen Aufbau ähnlich und bieten je nach Anforderung unterschiedliche modular implementierte Erweiterungen: so erfordert die Eintragung eines Objekts das Perl-Segment[6] *DCMI*; die Eintragung ei-

---

[1] MySQL kennt Fremdschlüssel nur für den Tabellentyp InnoDB. Für den hier verwendeten Tabellentyp MyISAM bietet sich statt dessen der SQL-Befehl JOIN an.

[2] Zusätzlich wurde die Programmiersprache JavaScript für die Programmierung von Metatag- und Template-Generatoren der *datenbankbasierten Entwicklungsumgebung* verwendet.

[3] Hiermit sind die Prozesse *Eintragung*, *Bearbeitung* und *Recherche* von Objekten/Modulen gemeint.

[4] Programmausführende Prozesse kommen (noch) nicht vor, lassen sich aber für künftige Erweiterungen berücksichtigen.

[5] Objekt-Archive liegen als zip-Dateien vor, wobei zu jedem Objekt zusätzlich ein externer Beschreibungssatz als rdf-Datei existieren muss.

[6] Perl-Segmente bezeichnen hier Perl-orientierte Programmteile.

ner Lerneinheit erfordert zusätzlich das Perl-Segment *A-F-Matrix*; die Eintragung eines Lehrgangs erfordert neben den beiden vorgenannten Perl-Segmenten noch das Perl-Segment *Lernziele*.

Das Perl-Segment ***DCMI*** [Quellcode (DCMI)] wiederum ist eines besseren Verständnisses wegen, und um den Nutzer nach Möglichkeit nicht direkt mit dem u.U. ungewohnten DC-Standard zu konfrontieren, in drei Bereiche mit zugeordneten Inhalten unterteilt [Kapitel 4.1.1 Standardisierung]:

Referentieller Bereich (Inhaltsorientierte Kriterien):
Nutzername, Passwort, Thema, Zuordnung, Beschreibung, Keywords, Mitarbeiter, Herausgeber, Quellen, Sprache, Copyright

Relationaler Bereich (Bezugsorientierte Kriterien):
Genre, Hyperlinks, Bezug

Genereller Bereich (Dateiorientierte Kriterien):
Objektname bzw. Modulname

Das Perl-Segment ***A-F-Matrix*** [Quellcode (DCMI, A-F-Matrix)] enthält die folgenden zwei Bereiche mit den nach G. Ropohl zugeordneten Inhalten [Kapitel 4.1.1 Standardisierung].

Anwendungsfelder
Versorgung und Entsorgung, Transport und Verkehr, Information und Kommunikation, Automation [Tabelle 5: Attribute-Funktionen-Matrix (A-F-Matrix)]

Technische Sachsysteme
Systeme der Wandlungs-, Übertragungs- und Speicherungstechnik [Tabelle 4: Klassifikation technischer Sachsysteme]

Das Perl-Segment ***Lernziele*** [Quellcode (DCMI, A-F-Matrix, Lernziele)] stellt den Bezug zu den Ausführungen über Lernziele für drei Bereiche mit den jeweils zugeordneten Inhalten her [Kapitel 3.1 Multimediales Lernen].

Lernzielorientierung
Gegenstand, Thematik

Lernzieltaxonomie
Kognitive, affektive und psychomotorische Dimension

Schwerpunktlernziel
Schwerpunktziel, Feinziele

Die folgende Abbildung 13 soll, ausgehend vom graphischen Frontend, den modularen Aufbau der am Vorgang des aktiven Datenbank-Zugriffs beteiligten Perl-Segmente *DCMI*, *A-F-Matrix* und *Lernziele* verdeutlichen.

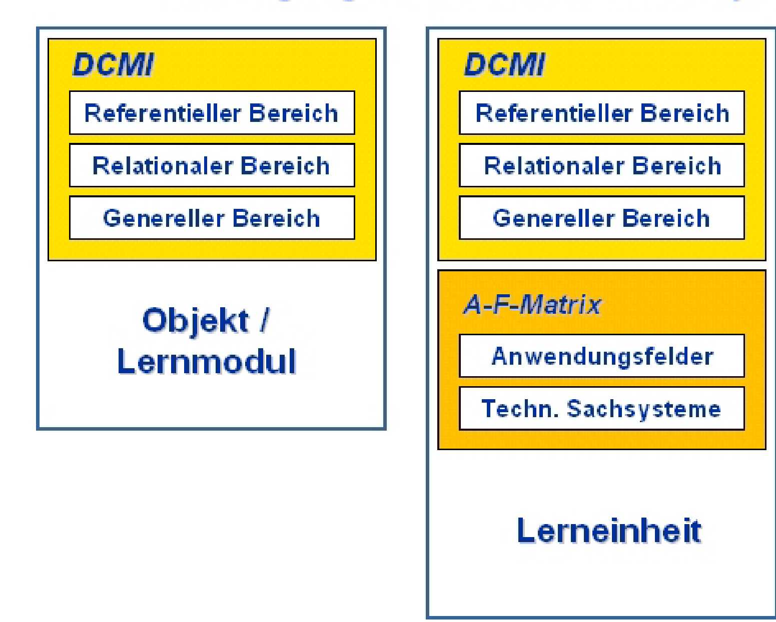




Abbildung 13: Modularer Aufbau aktiver Datenbank-Zugriffe

Auch bei der Korrektur von Objekten/Modulen bildet ein ähnlicher modularer Aufbau die Grundlage von vorzunehmenden Veränderungen.

Passive Datenbank-Zugriffe:

- Abfrage von Objekten
- Abfrage von Lernmodulen, Lerneinheiten und Lehrgängen
- Abfrage von Objekt- und Modulinformationen

Ein passiver Datenbank-Zugriff beschränkt sich ausschließlich auf den Prozess *Abfrage von Objekten/Modulen*. Die damit verbundenen durch PHP-Programme realisierten Nutzerschnittstellen sind ebenfalls in ihrem graphischen Aufbau ähnlich und bieten je nach Anforderung auch unterschiedliche modular implementierte Erweiterungen: so erfordert die Abfrage eines Objekts eine Eingabe über das PHP-Segment[1] *Angabe von Suchbegriffen*; die Abfrage einer Lerneinheit erfordert zusätzlich eine Eingabe über das PHP-Segment *Auswahl innerhalb der A-F-Matrix*; die Abfrage eines Lehrgangs erfordert neben einer Eintragung in den beiden vorgenannten

[1] PHP-Segmente bezeichnen hier PHP-orientierte Programmteile

PHP-Segmenten noch eine Eintragung in das PHP-Segment *Abfrage von Lernzielen* (Abbildung 14).

Die Generierung von Lernobjekten ist in diesem Zusammenhang sehr eng mit den Möglichkeiten einer Recherche verbunden. Um beispielsweise ein Lernmodul zu generieren, ist ebenso eine Objekt-Recherche notwendig, wie bei der Generierung einer Lerneinheit.

Abbildung 14 gibt die vier wesentlichen durch graphische Frontends realisierten Möglichkeiten passiver Datenbank-Zugriffe mit ihrem jeweiligen modularen Aufbau[1] und den zugehörigen Ergebnissen[2] wieder. Sie werden im folgenden erläutert.

Das PHP-Segment **Angabe von Suchbegriffen** [Quellcode (Suchbegriff)] stellt ein Eingabefeld für frei formulierte Suchbegriffe zur Verfügung und sendet diese in Form einer Anfrage an die Datenbank *mmdb* weiter. Das Resultat ist im Falle der Generierung eines Lernmoduls eine ungeordnete Ergebnisliste verfügbarer Objekte, während es im Falle der Generierung einer Lerneinheit eine geordnete und verlinkte Ergebnisliste verfügbarer Objekte ist.

Das PHP-Segment **Auswahl innerhalb der A-F-Matrix** [Quellcode (Suchbegriff, A-F-Matrix)] stellt über Auswahllisten die o.g. Anwendungsfelder und die z. Zt. benannten technischen Sachsysteme zur Verfügung. Gefolgt von einer so eingeschränkten Angabe von Suchbegriffen ist das Resultat eine geordnete Ergebnisliste verfügbarer Lerneinheiten. In den Auswahllisten für Anwendungsfelder und Technische Sachsysteme werden nur diejenigen Inhalte gelistet, die auch vorhanden und somit durch die Datenbank *mmdb* abrufbar sind.

Das PHP-Segment **Abfrage von Lernzielen** [Quellcode (Suchbegriff, A-F-Matrix, Lernziel)] schränkt die Abfrage an die Datenbank *mmdb* zusätzlich stark ein: nach einer Auswahl innerhalb der A-F-Matrix und vor einer Angabe von Suchbegriffen müssen Lernziele in den Bereichen Lernzielorientierung, Lernzieltaxonomie und Schwerpunktlernziel angegeben werden. Falls für diese Bereiche keine oder nur sehr wenige Datensätze existieren, wird der jeweilige Nutzer darauf hingewiesen. Als Ergebnis wird dann eine geordnete Liste der verfügbaren Lehrgänge ausgegeben.

---

[1] Die Recherche von Lernmodulen ist in Abbildung 14 nicht dargestellt. Sie erfolgt, genau wie die Objekt-Recherche, lediglich durch eine Angabe von Suchbegriffen.

[2] Mit Ausnahme der 'Generierung von Lernmodulen' erfolgt die Ausgabe einer Ergebnisliste immer in geordneter Form. Diese Form wird durch die Häufigkeit des Auftretens von zu recherchierenden Schlüsselwörtern bestimmt.

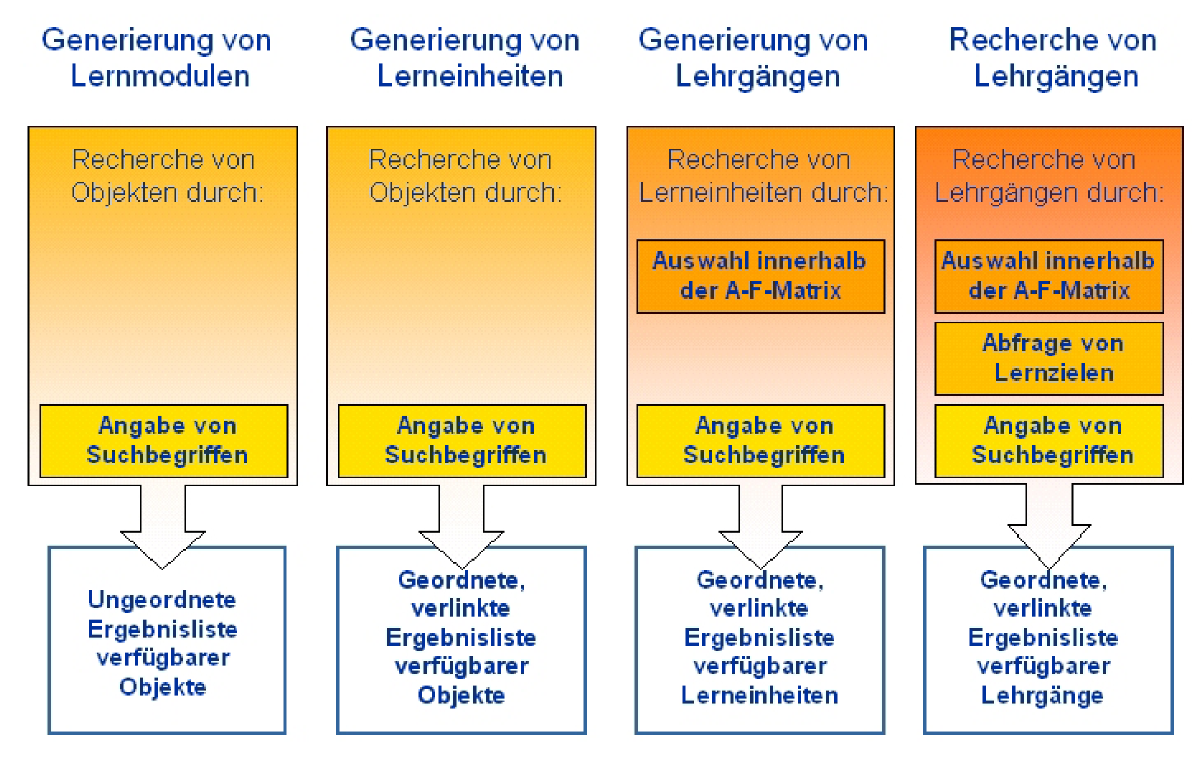


Abbildung 14: Modularer Aufbau passiver Datenbank-Zugriffe

Die Abfrage von Objekten/Modulen ist mit der Generierung von Lernobjekten verknüpft. Aus diesem Grund existiert keine explizit ausgeführte Möglichkeit einer Recherche von Lerneinheiten, da diese bei der Generierung von Lehrgängen integriert ist [Kapitel 4.2.4 Nutzerschnittstelle INTEGER].

Bewertbarkeit

Ein weiteres Element des modularen Ansatzes ist die **modulare Kopplung**, welche die Möglichkeit einer Bewertung vorhandener Objekte/Module bietet. So sind sämtliche durch die Datenbank *mmdb* zur Verfügung stehenden Objekte/Module online bewertbar: der Zusatz der Bewertbarkeit ist datenbankbasiert und untrennbar mit den Inhalten der Datenbank *mmdb* verbunden[1]. Der Urheber eines Objekts/Moduls wird dadurch angehalten, die fachwissenschaftliche Korrektheit seiner Angaben zu beachten. Die nachfolgende Abbildung 15 zeigt beispielhaft als Objekt die graphische Visualisierung einer Form des Snelliusschen Brechungsgesetzes mit der damit verbundenen Objekt-Bewertung.

Programmtechnisch wird eine Frame-Struktur erzeugt, die im oberen Frame die Möglichkeit der Bewertung bietet und im unteren Frame das zu bewertende Objekt/Modul

[1] Zu diesem Zweck existiert eine eigene Entitätsmenge.

selbst zeigt. Bewertet wird ein Objekt/Modul durch die Aktivierung eines Auswahlfeldes, wobei der Grad der Nutzbarkeit durch die Vergabe einer Note von *1* bis *5* mit *hoch* über *mäßig* bis *gering* gewählt werden kann. Zusätzlich enthält dieses Frame Informationen über das *Thema*, die *Objekt/Modul-ID* und die *e-Mail* des *Urhebers*. Die durchschnittliche Bewertung des Objekts/Moduls kann erst nach Abgabe der eigenen Bewertung, also nach Betätigung des Bewerten-Buttons, eingesehen werden.

Diese einfache Form der Bewertung wurde gewählt, um der User-Community die Möglichkeit zu geben, ihre Objekte/Module in Eigenregie zu verwalten. Jeder registrierte Nutzer kann den Urheber von Objekten/Modulen via e-Mail auf eventuell vorzunehmende Korrekturen hinweisen. Nach Ablauf einer durch den Administrator der Datenbank *mmdb* vorgegebenen Zeit, werden alle Objekte/Module im Hinblick auf die vorgenommenen Bewertungen automatisch geprüft und gegebenenfalls aus der Datenbank *mmdb* entfernt.

Die User-Community ist dadurch angehalten, in selbstverantwortlicher und somit nachhaltiger Weise mit den Inhalten der *datenbankbasierten Entwicklungsumgebung* umzugehen. Sie übernimmt das Einbringen und die Aktualisierung der Datensätze und entscheidet letztlich über den Datenbestand.

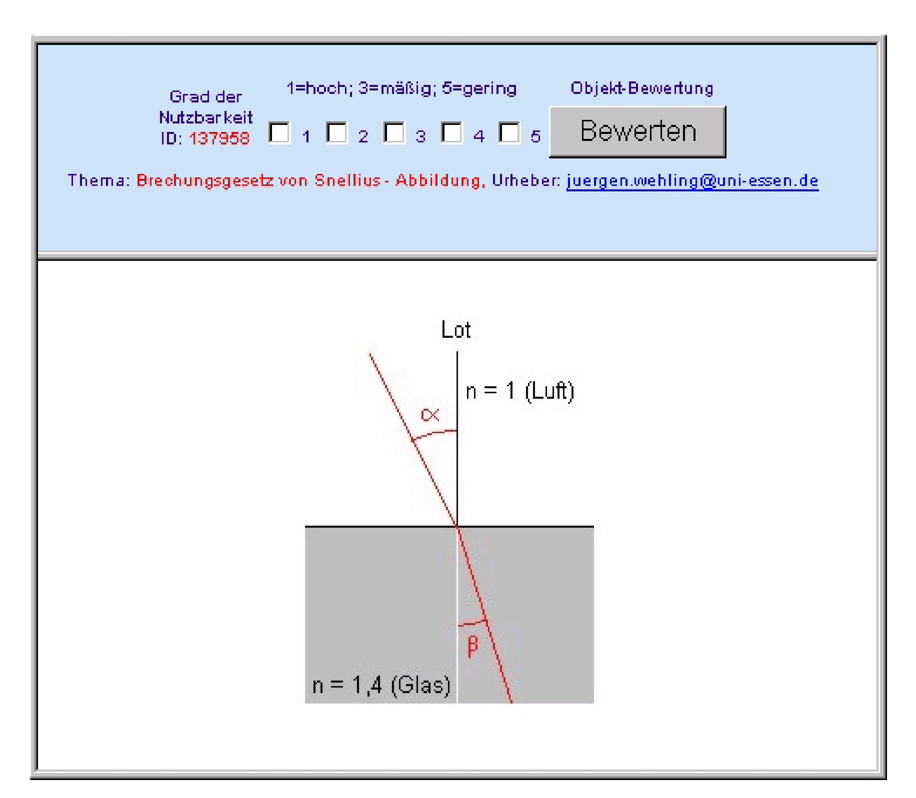


Abbildung 15: Modulare Kopplung von Objekt und Objekt-Bewertung

Generatoren und Templates

Die *datenbankbasierte Entwicklungsumgebung* bietet dem Nutzer zwei Metatag-Generatoren[1] auf der Basis von JavaScript, um Daten für eine Eintragung in die Datenbank *mmdb* zu generieren. Diese Generatoren erzeugen Metatag-basierte Datensätze für **alle** Objekte, die sich in die Datenbank *mmdb* eintragen lassen. Folgende Generatoren sind verfügbar:

- Generator für DC-Metatags für alle Objekte
- Generator für DC-Metatags nach RDF für alle Objekte.

Die Implementierung dieser Generatoren wurde nach Hinweisen der User-Community umgesetzt. So sollte allen Nutzern die Möglichkeit gegeben werden, Objekte in einer einheitlichen Vorgehensweise und in standardisierter Form einzutragen. Zusätzlich sollte ihnen auf diese Weise die Vorbereitung der Objekte für eine Eintragung in die Datenbank *mmdb* erleichtert werden. Diese Generatoren vermitteln dem Nutzer einen einfachen und leicht verständlichen Zugang zu einer strukturierten Ablage von Objekten unter gleichzeitiger Berücksichtigung des DC-Standards.

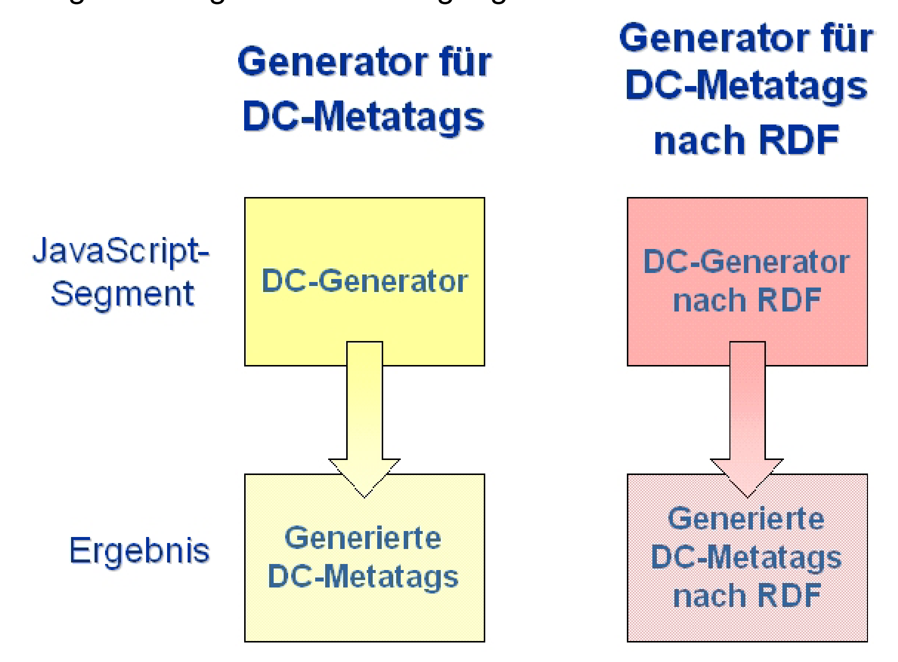


Abbildung 16: Prinzipieller Aufbau der Generatoren für Metatags nach DC

Neben diesen Metatag-Generatoren wurden ebenfalls auf der Basis von JavaScript Template-Generatoren entwickelt, die in ihrer grundsätzlichen Funktionsweise auf den vorgenannten Generatoren aufbauen.

- Template-Generator mit DC-Metatags für htm/html-Objekte
- Template-Generator mit DC-Metatags nach RDF für htm/html-Objekte

[1] Zusätzlich wurde noch ein dritter Generator für konventionelle Metatags implementiert, der jedoch nur für internetbasierte Hauptsuchmaschinen Relevanz hat.

Während Metatag-Generatoren alle Objekte für eine Eintragung in die Datenbank *mmdb* vorbereiten können, wurden die Template-Generatoren ausschließlich für die Eintragung von Objekten im **htm/html**-Format geschaffen. Dieser Schritt erwies sich als notwendig und sinnvoll, da viele Nutzer vorwiegend textbasierte Objekte in die Datenbank *mmdb* eintrugen, dabei aber möglichst gar nicht mit (X)HTML-Konstrukten konfrontiert werden wollten.

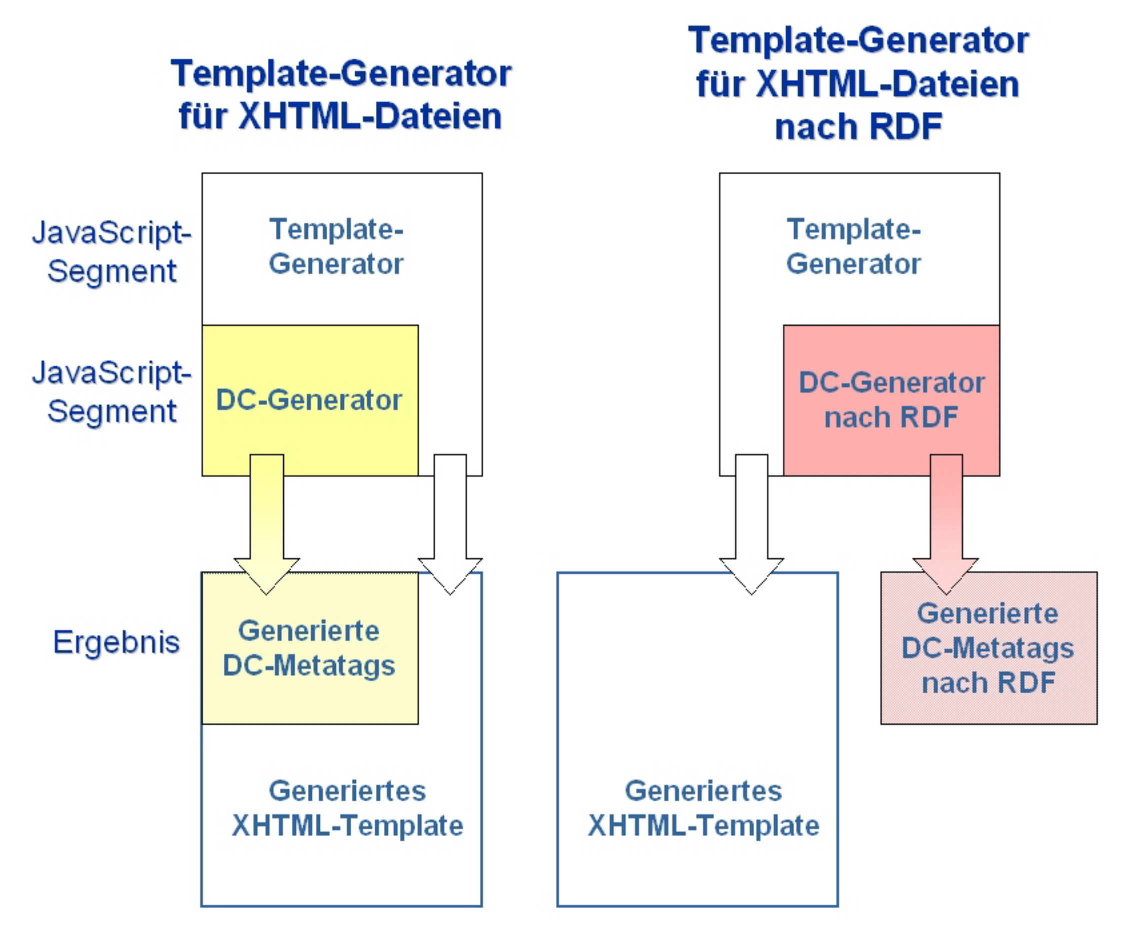


Abbildung 17: Modularer Aufbau der Template-Generatoren nach XHTML

Diese beiden in ihrem modularen Aufbau durch Abbildung 17 dargestellten Template-Generatoren wurden aus JavaScript-Segmenten, die schon für die Realisierung der Metatag-Generatoren nach DC und RDF [Abbildung 16: Prinzipieller Aufbau der Generatoren für Metatags nach DC] entwickelt wurden, aufgebaut und durch zusätzliche Elemente, welche die speziell auf htm/html-basierte Objekte zugeschnitten Ergebnisse wiedergeben, ergänzt. Zusätzliche Elemente sind in diesem Fall der Einsatz von CSS und die Implementierung korrekter XHTML-Syntax im Aufbau des Templates.

Zusammenfassend lässt sich Kapitel 4.1 damit schließen, dass schon allein durch die im modularen Ansatz realisierten Standardisierungs-Elemente wesentliche in dieser Arbeit weiter vorn genannte konzeptionelle Forderungen [Kapitel 3.2.1 Ansatzpunkt] erfüllt werden:

- Durch die modulare Kopplung von Objekt/Modul und Bewertbarkeit bietet die Datenbank *mmdb* der User-Community die Möglichkeit einer Wartung im fachwissenschaftlichen Sinne.

- Durch den konsequenten Einsatz freier Software und allgemein anerkannter standardisierter Elemente (offene Standards, Linux-Kernels) sind wesentliche Voraussetzungen für die Eigenschaften der Interoperabilität und Portierbarkeit gegeben.

4.2 Datenbankbasierte Entwicklungsumgebung

Die **INT**egrierte **E**ntwicklungsumgebung für eine **GE**nerierung von Le**R**nobjekten INTEGER und die **M**ulti**M**edia**D**aten**B**ank für den **T**echnik**U**nterricht MMDB-TU bilden eine fest verbundene Einheit, nämlich die *datenbankbasierte Entwicklungsumgebung*. Die Trennung dieser symbiotischen Einheit durch zwei unterschiedliche Begriffe findet seine Begründung in der Stufe 2 der ganzheitlichen Konzeption eines multimedialen Lernens: **Strukturiertes Ablegen von Objekten** sowie neu **strukturiertes Kombinieren von Objekten zu Lernobjekten** (Lernmodule, Lerneinheiten, Lehrgänge) [Kapitel 3.2 Forderungen].

4.2.1 Entscheidungsbegründung

Das **Strukturierte Ablegen von Objekten** kann nur auf der Grundlage eines RDBMS geschehen. Genau aus diesem Grunde wurde auf der Basis von MySQL die Datenbank *mmdb* entwickelt, welche selbst wiederum nur über die graphisch orientierte Nutzerschnittstelle MMDB-TU den **aktiven Datenbank-Zugriff** erlaubt. Die Browser-konforme Umsetzung dieser Schnittstelle ist durch den Einsatz von Perl realisiert worden. Dabei wurden im wesentlichen die Perl-Module CGI [Stein, 1998] und DBI [Descartes, 2001] eingesetzt. Über das Modul *cgi.pm* wird die graphische Benutzeroberfläche realisiert, während das Modul *dbi.pm* Verbindungen zur Datenbank *mmdb* im Rahmen aktiver Datenbank-Zugriffe herstellt.

Für das **Strukturierte Kombinieren von Objekten zu Lernobjekten** wurde INTEGER konzipiert. INTEGER wird als Nutzerschnittstelle ausschließlich für **passive Datenbank-Zugriffe** eingesetzt und stellt auf der Basis der Datenbank *mmdb* in Einheit mit der Nutzerschnittstelle MMDB-TU schließlich die *datenbankbasierte Entwicklungsumgebung* dar. Diese durch PHP realisierte Schnittstelle nutzt z. Zt. weder Session-IDs[1] noch Cookies, was sich aber jederzeit bedarfsbedingt ändern lässt. In diesem Zusammenhang erwies sich beispielsweise die Annahme von Cookies zur Wiederherstellung vergangener Sessions als problematisch, da diese durch den jeweiligen Nutzer erst explizit im Konfigurationsmenü seines Browsers freigegeben werden muss.

Wie gerade dargelegt, nimmt die Datenbank *mmdb* eine zentrale Position ein, da sie den Nutzerschnittstellen MMDB-TU und INTEGER gleichermaßen zur Verfügung steht. Die nachstehende Abbildung 18 visualisiert diesen Zusammenhang.

[1] Eine Authentifizierung des jeweiligen Nutzers erfolgt durch die Datenbank *mmdb*.

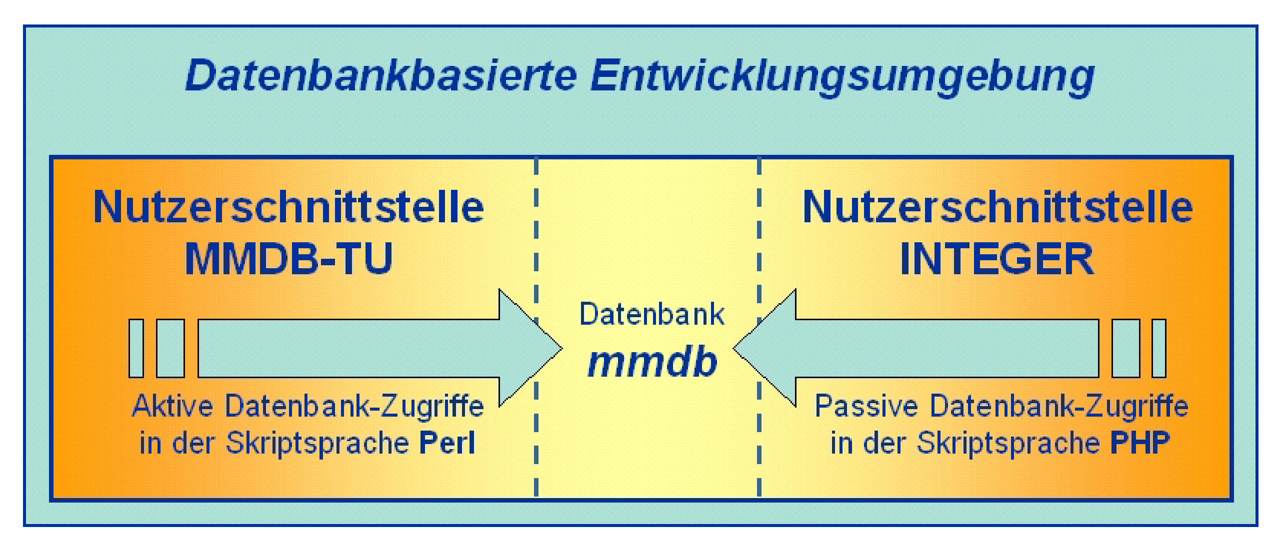


Abbildung 18: Die *datenbankbasierte Entwicklungsumgebung*

Um die beiden durch die Skriptsprachen Perl und PHP aufgebauten Nutzerschnittstellen wurde ein **Hilfe-** bzw. **Informationssystem** aufgebaut, das jedem registriertem Nutzer per Mausklick zur Verfügung steht.

Bei einer Eintragung von Objekten/Modulen kann für jede durch den Nutzer vorzunehmende Eintragung ein **Hilfefenster** mit näheren Erläuterungen und Beispielen eingeblendet werden, während bei einer Abfrage von Objekten/Modulen **Informationsfenster** abgerufen werden können [Abbildung 31: Screenshot: Objekt-Information], welche die Objekte/Module visualisieren, wobei gleichzeitig die Möglichkeit einer Bewertung angeboten wird [Abbildung 15: Modulare Kopplung von Objekt und Objekt-Bewertung] und zusätzlich der Datenbeschreibungssatz des Objekts abrufbar ist. In jedem Fall wird die Programmiersprache JavaScript eingesetzt, d. h.: sowohl Perl- als auch PHP-Programme enthalten integrierten JavaScript-Code.

Der Einsatz von JavaScript bietet zusätzlich zur Schaffung eines zeitgemäßen Webdesigns in Form von Fenstertechniken für das Hilfe- und Informationssystem die Möglichkeit, sicherheitstechnische Aspekte zu ergänzen [Kapitel 4.2.4 Sicherheitsaspekte].

Das Auswahlmenü der *datenbankbasierten Entwicklungsumgebung* stellte eine besondere Herausforderung für die in den vorhergehenden Kapiteln geforderte Browserunabhängigkeit dar. Ursprünglich als reines JavaScript-Programm ausgeführt, war das Auswahlmenü unter den Browsern Internet Explorer 4 und 5, Netscape 4.x und Opera 5 voll funktionsfähig. Mit Aufkommen der höheren Versionen dieser Browser war jedoch die Forderung einer Browserunabhängigkeit nicht mehr zu erfüllen, da die wichtigsten Hersteller offensichtlich immer mehr proprietäre Standards implementieren: sogar in Java erstellte Programme liefern abweichende Ergebnisse[1].

[1] Der bedeutendste Nachteil des Internet Explorers 6 ist beispielsweise die Verwendung einer veralteten Virtual Machine. Damit fehlt eine Unterstützung von Java 2 Applets, womit künftige Probleme bei der Wiedergabe von Java-Applets vorprogrammiert sind. Zusätzlich werden Netscape-Plug-ins nicht mehr unterstützt!

Die logische Konsequenz bestand darin, den beliebtesten Browser zu bevorzugen. Hierbei zeigt die nachstehende Abbildung 19 sehr deutlich, dass der Internet Explorer als Browser am häufigsten eingesetzt wird und durchaus interessante Alternativen wie Netscape/Mozilla leider nur verschwindend geringe Anteile aufweisen.

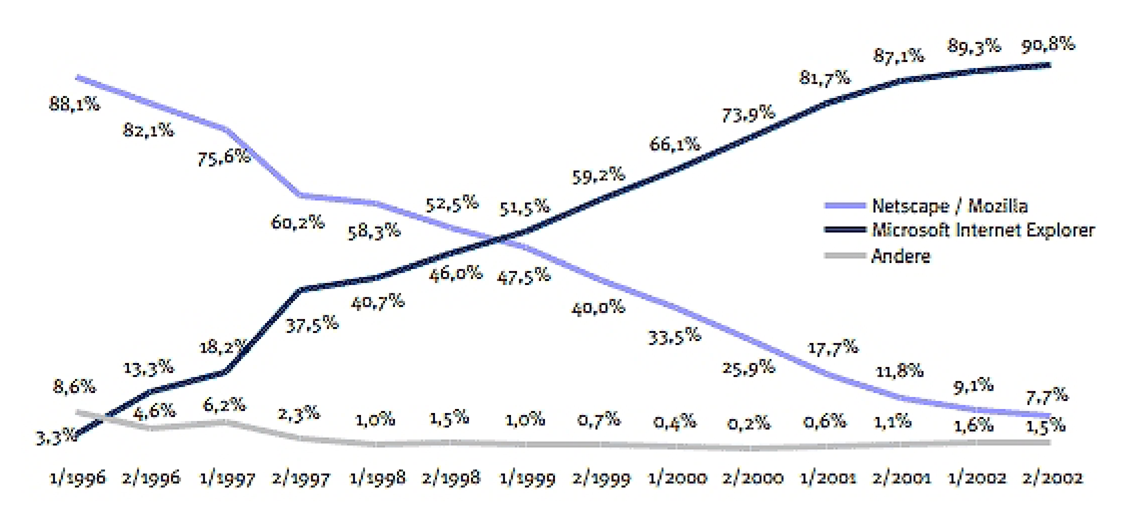


Abbildung 19: Analyse der Browser-Anteile [Fittkau, 2003]

Es ist anzumerken, dass in Abbildung 19 der Internet Explorer in der Version 5 (15,8 %), 5.5 (30,2 %) und 6 (44 %) zusammengefasst dargestellt ist und die Versionen 1.x bis 4.x lediglich 0,8 % ausmachen. Aufgrund seines Anteils von 90,8 % wird der Internet Explorer ab Version 5 für ein Arbeiten mit der *datenbankbasierten Entwicklungsumgebung* empfohlen. Die Kompatibilität zu den übrigen Browsern wird in regelmäßigen Abständen geprüft, hat jedoch geringere Priorität; insbesondere der mittlerweile lizenzpflichtige Browser Opera wird nicht weiter unterstützt.

Bedingt durch den stark expandierenden Markt für TFT-LCDs ist bei CRT-Monitoren ein starker Preisverfall festzustellen, der sich indirekt auf das graphische Erscheinungsbild der *datenbankbasierten Entwicklungsumgebung* auswirkt. Änfanglich für eine Auflösung von 800x600 Pixel programmiert, wird mittlerweile eine Auflösung von mindestens 1024x768 Pixel empfohlen. Insbesondere bei sehr vielen aktivierten Fenstern des Hilfe- und Informationssystems kann sich eine noch höhere Bildschirmauflösung (bis 1600x1200 Pixel) sehr vorteilhaft auf das Arbeiten mit der *datenbankbasierten Entwicklungsumgebung* auswirken. Die nachfolgende Abbildung 20 zeigt die Startseite der deutschsprachigen Version in der Auflösung 1024x768 Pixel.



Abbildung 20: Frames der *datenbankbasierten Entwicklungsumgebung*

Der Bildschirm ist prinzipiell in drei Bereiche (stilisiert dargestellt als Frames in Abbildung 20) geteilt: den **Header** mit den graphischen Elementen *MultiMediaDaten-Bank* und *INTEGER*, dem **Menü** und dem **Hauptbildschirm**, der die Arbeitsoberfläche der *datenbankbasierten Entwicklungsumgebung* darstellt. Die Frame-Darstellung von Header und Menü ist in jedem Fall durchgängig präsent, wobei die Höhe des Headers (93 Pixel) sowie die Breite des Auswahlmenüs (133 Pixel) unabhängig von der gewählten Bildschirmauflösung ist und daher immer gleich bleibt. Diese Frame-Darstellung wurde aus ergonomischen Gründen gewählt: bei minimalen Abstrichen an die Frames Header und Menü ergibt sich ein möglichst großer nutzbarer Arbeitsbereich im Hauptbildschirm.

4.2.2 Datenbank mmdb

Wie die meisten relationalen Datenbanken ist auch die Datenbank *mmdb* durch Tabellen mit jeweils definierten Primärschlüsseln aufgebaut, wobei als Tabellentyp der von MySQL standardmäßig vorgegebene MyISAM[1] gewählt wurde. Eine Alternative bietet der Tabellentyp BDB[2], der, ähnlich dem Tabellentyp InnoDB[3], Transaktionen unter-

[1] ISAM ist die Abkürzung für **I**ndexed **S**equential **A**ccess **M**ethod (indexsequentieller Zugriff). Dieser Tabellentyp dient einer schnellen Ermittlung eines Datensatzes in umfangreichen Datenbanken.

[2] BDB ist die Abkürzung für Berkeley DataBase.

[3] InnoDB wurde von Heikki Tuuri geschrieben und leitet sich vom damit verbundenen Firmennamen **Inno** her.

stützt. Für die Datenbank *mmdb* reicht jedoch der Tabellentyp MyISAM vollkommen aus [Kapitel 4.1.3 Integration].

In der Anfangsphase unter dem Betriebssystem SuSE Linux 7.1 Professional bestand die *mmdb* aus den in Abbildung 21 wiedergegebenen 7 Tabellen, die in noch nicht normalisierter Form vorlagen, jedoch ein erstes Arbeiten mit der *datenbankbasierten Entwicklungsumgebung* möglich machten.

table	Action						Records
af_matrix	Browse	Select	Insert	Properties	Drop	Empty	15
basismodule	Browse	Select	Insert	Properties	Drop	Empty	11
hypermodule	Browse	Select	Insert	Properties	Drop	Empty	1
lernziele	Browse	Select	Insert	Properties	Drop	Empty	1
nutzer	Browse	Select	Insert	Properties	Drop	Empty	25
obermodule	Browse	Select	Insert	Properties	Drop	Empty	7
objekte	Browse	Select	Insert	Properties	Drop	Empty	22

Abbildung 21: Entitäten der ursprünglichen Datenbank *mmdb*

Zum damaligen Zeitpunkt existierten nur wenige Datensätze und auch die Anzahl der Nutzer war gering. Lernmodule sind unter dem Tabellennamen *basismodule* eingetragen, Lerneinheiten und Lehrgänge unter *obermodule* bzw. *hypermodule*. Im Zuge der laufenden Modifizierung der Datenbank *mmdb* wurden Änderungen notwendig, die insbesondere aufgrund des erhöhten Nutzeraufkommens und der damit verbundenen Ansprüche einen Umbau mit darauf folgenden Formen der Normalisierung notwendig machten. Wesentliche konzeptionelle Elemente der anfänglichen Datenbank *mmdb* wurden jedoch beibehalten. Abbildung 22 zeigt in Form eines ER-Diagramms die logische und physikalische Struktur der jetzigen Datenbank *mmdb* unter dem Betriebssystem SuSE Linux 8.1 Professional.

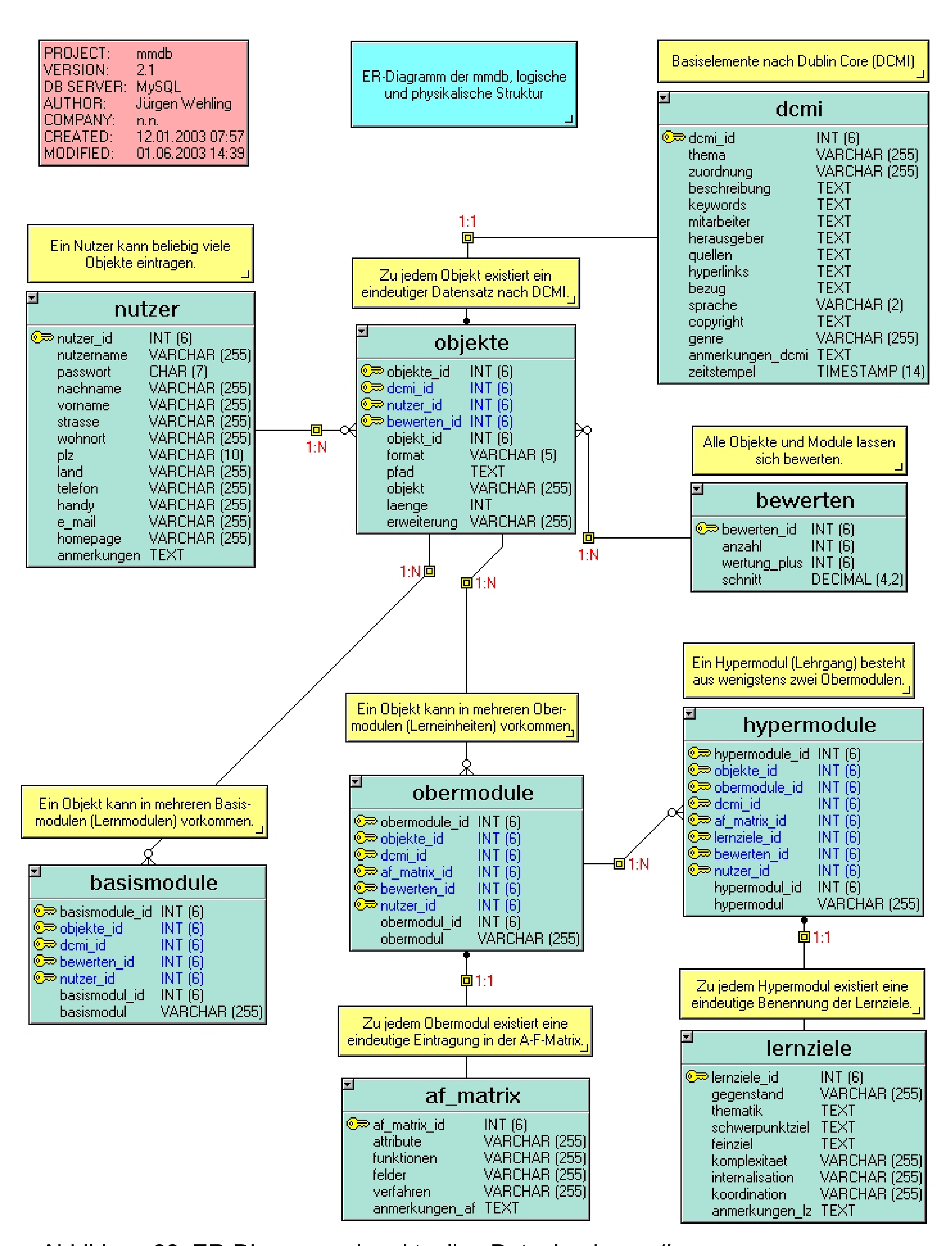


Abbildung 22: ER-Diagramm der aktuellen Datenbank *mmdb*

Wie Abbildung 22 zeigt, liegt die jetzige Datenbank *mmdb* in der 3. Normalform vor, d. h.: alle Entitäten, die nicht in unmittelbarer Abhängigkeit zum Primärschlüssel der entsprechenden Entitätsmenge stehen, besitzen eine eigene Entitätsmenge. Eingetragen sind auch die Relationen zwischen den einzelnen Entitätsmengen. So existiert beispielsweise eine 1:N Relation zwischen den Entitätsmengen *objekte* und *basismodule*, d. h. ein Objekt kann in mehreren Basismodulen (Lernmodulen) vorkommen; zwischen den Entitätsmengen *dcmi* und *objekte* existiert eine 1:1 Relation, d. h. zu jedem Objekt existiert ein eindeutiger Datenbeschreibungssatz nach DC.

Die Datenbank *mmdb* muss in geeigneter Weise administrierbar sein. Als wesentliches Instrumentarium wird die Open-Source-Software phpMyAdmin [phpMyAdmin, 2003] eingesetzt, die z. Zt. in der Version 2.4.0 vorliegt [Abbildung 35: Screenshot: Administrativer Bereich, phpMyAdmin]. Neben dieser als PHP-Programm ausgeführten Benutzeroberfläche für das RDBMS MySQL sind zusätzlich spezielle Programme zur Datenbank-Verwaltung sowohl für die Nutzer-Ebene, als auch für die Administrator-Ebene notwendig. Sämtliche Programme der Nutzer-Ebene zur Verwaltung nutzerspezifischer Datensätze werden im folgenden Kapitel beschrieben [Kapitel 4.2.3 Nutzerschnittstelle MMDB-TU]. Diejenigen Programme, die für die administrative Ebene notwendig sind, befinden sich in einem späteren Kapitel dieser Arbeit [Kapitel 4.2.5 Administrativer Bereich].

4.2.3 Nutzerschnittstelle MMDB-TU

Das ER-Diagramm in Abbildung 22 zeigt, dass der zentrale Bestandteil der Datenbank *mmdb* das Objekt selbst ist. Zur Verdeutlichung der zur Realisierung der Nutzerschnittstelle MMDB-TU arbeitenden Programme wird das Beispiel des Eintrags einer XHTML-Datei in die Datenbank *mmdb* in exemplarischer Form ausgewählt. Eine Eintragung dieser Art erfolgt durch das Perl-Programm *header_parsing.pl* [Abbildung 23: Programm-Flussbild header_parsing.pl]. Es bezieht sich beispielhaft auf das Kapitel 4.1.2 [Konzeption] und die dort wiedergegebenen Listings [Abbildung 5: Implementierung konventioneller Metatags nach DC] und zerlegt den Header einer XHTML-Datei automatisch in die einzelnen Elemente, die für eine Einordnung nach dem DC-Standard in die Datenbank *mmdb* relevant sind. Zum besseren Verständnis dieses Vorgangs folgt hier eine Beschreibung ausgewählter Code-Segmente des genannten Perl-Programms.

Nach Festlegung der Variablen und Konstanten wird das Programm durch eine Einbindung der notwendigen Perl-Module initialisiert:

```
use CGI qw/:standard :html3/;
use DBI;
use MIME::Base64 ();
use File::Basename;
use Net::SMTP;
```

Nachdem Nutzernamen und Passwort des registrierten Nutzers erfolgreich verifiziert wurden, erfolgt die Ermittlung der zugehörigen `nutzer_id`:

```
$DSN = "DBI:mysql:mmdb";
$dbh = DBI->connect($DSN, 'root', '********')
       or die "Verbindungsaufnahme nicht moeglich";
       $sth = $dbh->prepare("SELECT nutzer_id FROM nutzer WHERE
       (nutzername = '$nutzername' AND passwort = '$passwort')");
       $sth->execute();
              while ( @result = $sth->fetchrow_array ){
              for ($i = 0; $i < @result; $i++) {
              $nutzerdaten[$i] = $result[$i];
              }
              };
       $sth->finish();
$dbh->disconnect;
```

Die `nutzer_id` legt den Pfad des jeweiligen Nutzers zu den eigenen abgespeicherten Objekten/Modulen mit den zugehörigen Lese-, Schreib- und Ausführungsrechten fest. Nachdem vom Programm ein gültiger Objektname und der MIME-Typ `text/html` erkannt wurde, erfolgt eine Zerlegung des kompletten, übermittelten Dateipfades mit dem Ziel, den eigentlichen Dateinamen des Objekts zu isolieren und auf eine schon mögliche Existenz innerhalb der Datenbank *mmdb* zu überprüfen. Weist das zu übermittelnde Objekt noch keinen Eintrag in der Datenbank *mmdb* auf, so wird es im zugehörigen Nutzer-Account gespeichert und dem eigentlichen Parsing-Vorgang unterzogen. Das folgende Code-Segment gibt exemplarisch das durch einen einfachen regulären Ausdruck bestimmte Parsing des DC-Elements `DC.Rights` wieder:

```
$rights_string = "<META NAME=\"DC.Rights\" CONTENT=\"";
.-.-.
if ($a =~ /$rights_string/i)
       {$zeichenkette = substr($a, 32, 4096);
        entities_pruefen ();
        $zeichenkette =~ s/">//;
        $dc_rights = $zeichenkette;
       }
```

Die Funktion `entities_pruefen ()` untersucht den zu parsenden String auf HTML-spezifische Sonderzeichen[1] und ersetzt diese. Anschließend wird eine zufällige 6-stellige Objekt-ID erzeugt, die das einzutragende Objekt eindeutig identifiziert. Dabei wird geprüft, ob diese zufällig gewählte Objekt-ID existiert. In diesem Fall wird eine neue Objekt-ID erzeugt und diese wiederum überprüft:

[1] So wird beispielsweise "ä" durch ein "ä" ersetzt.

```
$DSN = "DBI:mysql:mmdb";
$dbh = DBI->connect($DSN, 'root', '')
       or die "Verbindungsaufnahme nicht moeglich";
       nochmal:
       $zufallsobjekt_id = int(rand(1000000));
       $sth = $dbh->prepare("SELECT objekt_id FROM objekte");
       $sth->execute();
       $match = "false";
              while ( @result = $sth->fetchrow_array ){
              if ($result[0] eq $zufallsobjekt_id){
              $match = "true" ;}
              };
       $sth->finish();
       if ($match eq "true") {
           goto nochmal;
           }
       elsif ($match eq "false") {
           $dbh->disconnect;
           };
```

Anschließend werden die Objekt-Daten für einen Abruf über INTEGER aufbereitet. Damit ist der Vorgang der schon angesprochenen modularen Kopplung gemeint [Abbildung 15: Modulare Kopplung von Objekt und Objekt-Bewertung]. An dieser Stelle soll lediglich die Generierung der damit verbundenen Steuerdatei wiedergegeben werden:

```
open (STEUERN, ">$steuern_objekt") or die "open failed: $!";
flock(STEUERN, 2);
seek(STEUERN, 0, 2);
print STEUERN "<html>\n";
print STEUERN "<head><title>\n";
print STEUERN "MMDB-TU: Objekt der MultiMediaDatenBank TechnikUnter-
richt\n";
print STEUERN "</title></head>\n";
print STEUERN "<frameset rows = \"120,*\">\n";
print STEUERN "<frame src = \"$bewerten_datei\" name=\"header\">\n";
print STEUERN "<frame src = \"$dateiname\" name=\"objekt\">\n";
print STEUERN "</frameset>\n";
print STEUERN "</html>\n";
flock(STEUERN, 8);
close (STEUERN);
```

Abschließend wird eine HTML-basierte Bestätigungsseite mit den geparsten und erzeugten Objekt-Daten ausgegeben und mit Hinweisen für eventuell fehlende oder unvollständige Einträge ergänzt. Nachdem das Objekt dann erfolgreich in die Datenbank *mmdb* übertragen wurde, erfolgt eine Bestätigung via E-Mail (hier auszugsweise wiedergegeben unter Nutzung des Mailservers des HRZ Essen):

```
my ($mailhost, $absender, $empfaenger, $smtp);
$mailhost = 'mailout.uni-essen.de';
$absender = 'webmaster@mmdb-tu.de';
$smtp = Net::SMTP->new($mailhost);
$smtp->mail($absender);
$smtp->to($e_mail);
$smtp->data();
$smtp->datasend(<<END);
To: $e_mail
Subject: Eintragung eines XHTML-Objekts in die MMDB-TU
Guten Tag $vorname $nachname,
die Eintragung eines HTML-Objekts in die MMDB-TU war erfolgreich.
.-.-.
Viel Erfolg bei der künftigen Nutzung unserer Datenbank.
$absender
END
$smtp->dataend();
$smtp->quit;
```

Damit ist der Parsing-Vorgang einer XHTML-Datei abgeschlossen. Es bleibt anzumerken, dass eine solche Datei, um Fehler zu vermeiden, möglichst mit Hilfe des Template-Generators für XHTML-Dateien erzeugt werden sollte [Abbildung 17: Modularer Aufbau der Template-Generatoren]. Weiterhin ist zu berücksichtigen, dass dieses Perl-Skript, wie alle anderen ebenfalls, über umfangreiche Routinen zum Abfangen von unzulässigen Angaben durch den Nutzer verfügt. Der komplette Quellcode kann unter http://www.mmdb-tu.de/programmcode/quellcodes.htm abgerufen werden, da allein dieses Perl-Skript einen Umfang von ca. 20 DIN A4 Seiten hat und daher an dieser Stelle nur auszugsweise dargestellt werden konnte.

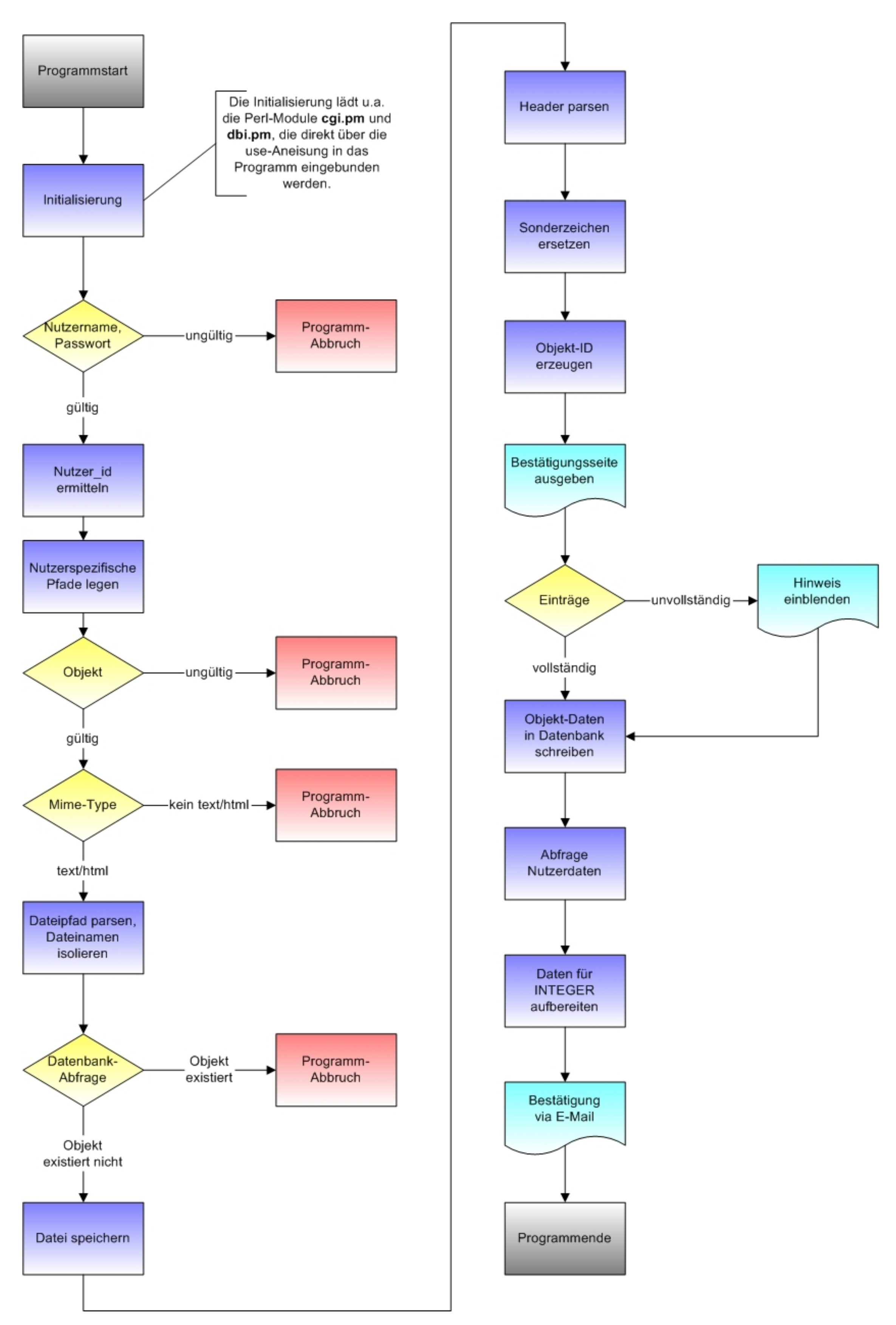


Abbildung 23: Programm-Flussbild *header_parsing.pl*

Für ein effektives Arbeiten über die Nutzerschnittstelle MMDB-TU ist das Zusammenspiel einer Vielzahl von Programmen notwendig. Die nachfolgende Abbildung 24 zeigt den dafür relevanten Auszug der Sitemap der *datenbankbasierten Entwicklungsumgebung* und stellt unter den jeweiligen Menüpunkten die zugehörigen Webpages (*.htm) bzw. Perl-Programme (*.pl) dar. Hierbei ist zu beachten, dass Programme, die dem Informationssystem zugeordnet sind, in PHP erstellt wurden (*.php4). Das Hilfesystem selbst ist aus Gründen der Übersichtlichkeit **nicht** aufgeführt!

Der Abbildung 24 ist weiterhin zu entnehmen, dass die Perl-Datei *header_parsing.pl* eine von insgesamt 12 Perl-Dateien ist, die alle ausschließlich für die weiter vorn schon beschriebenen aktiven Datenbank-Zugriffe erstellt wurden [Abbildung 13: Modularer Aufbau aktiver Datenbank-Zugriffe]. Davon gehören allerdings 4 Perl-Programme in den administrativen Bereich und sind dort beschrieben [Kapitel 4.2.5 Administrativer Bereich]. Bevor eine **Kurzbeschreibung der Perl-Programme** im einzelnen folgt, soll zuerst der Aufbau dieser Sitemap erläutert werden:

Ausgehend von der root (http://www.mmdb-tu.de und http://www.integer-tu.de), die als durchgezogene Linie im äußerst linken Bildteil dargestellt ist, findet sich der Menüpunkt *MMDB-TU* (1. Spalte[1]). Von hier zweigen die Untermenüs *Eintrag*, *Korrektur* und *Generatoren* ab (2. Spalte). Die in der 3. Spalte folgenden Menüpunkte sind nicht mehr Teil das Java-basierten Auswahlmenüs der *datenbankbasierten Entwicklungsumgebung*, sondern durch Hyperlinks in den zugehörigen Webpages realisiert. So finden sich beispielsweise in der Webpage *mmdbobjekte.htm* vier Möglichkeiten in Form von Hyperlinks, um ein Objekt in die Datenbank *mmdb* einzutragen [Abbildung 30: Screenshot: Eintragung eines Objekts]: *objekte.pl*, *header_parsing.pl*, *rdf_parsing.pl* und *zip_parsing.pl* (4. Spalte). Genau wie die Webpage mmdbkorrektur.htm, die ebenfalls vier weiterführende Hyperlinks enthält, bietet die Webpage *generatoren.htm* auch vier Möglichkeiten in Form von Hyperlinks, um die Javascript-basierten Metatag- und Template-Generatoren der MMDB-TU zu nutzen.

[1] Die Anordnung der Spalten 1-4 verläuft von links nach rechts.

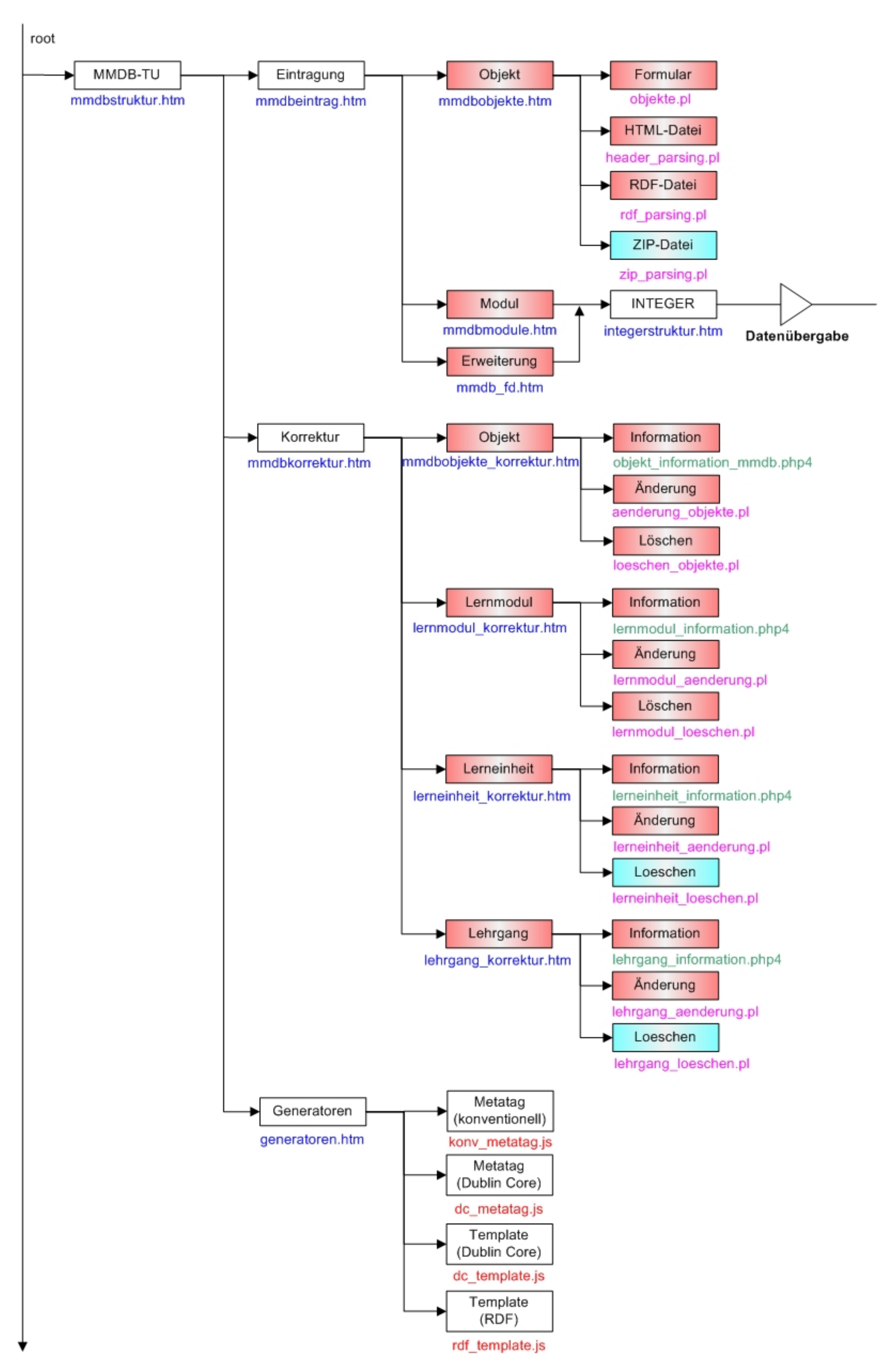


Abbildung 24: Sitemap der Nutzerschnittstelle MMDB-TU

Kurzbeschreibung der Perl-Programme:

Allen im folgenden beschriebenen Perl-Programmen ist gemeinsam, dass der Nutzer nach durchgeführtem aktivem Datenbank-Zugriff eine dynamisch generierte Webpage als Bestätigung sowie eine e-Mail mit allen zum Objekt/Modul erfolgten Eintragungen bzw. Änderungen erhält. Die zugehörigen Struktogramme und der komplette Quellcode lassen sich unter http://www.mmdb-tu.de/programmcode/quellcodes.htm abrufen.

objekte.pl: Dieses Programm stellt unter Nutzung des Perl-Moduls *cgi.pm* ein Formular für die Eintragung von Objekten nach dem DC-Standard zur Verfügung. Das Formular selbst weist 15 Felder in den Bereichen *referentiell*, *relational* und *generell* aus, die von jeweiligen Nutzer ausgefüllt werden müssen [Abbildung 13: Modularer Aufbau aktiver Datenbank-Zugriffe]. Jedes Feld verfügt über eine integrierte Hilfefunktion mit Beispielen für eine syntaktisch richtige Eintragung.

header_parsing.pl: Dieses Programm wurde weiter oben ausführlich beschrieben.

rdf_parsing.pl: Dieses Programm erfordert für das einzutragende Objekt zwei Dateien: eine Objekt-Datei selbst und eine externe Datei mit dem zugehörigen Beschreibungssatz [Abbildung 6: Medientyp murray2.class mit externer Datei murray2.rdf]. Zumindest aus syntaktischen Gründen sollte der Nutzer in diesem Fall einen der Generatoren der MMDB-TU nutzen.

zip_parsing.pl:. Dieses Programm wird im administrativen Bereich ausführlich beschrieben [Kapitel 4.2.5 Administrativer Bereich].

aenderung_objekte.pl, lernmodul_aenderung.pl, lerneinheit_aenderung.pl und lehrgang_aenderung.pl: Diese Programme sollen dem jeweiligen Nutzer helfen, Korrekturen an den vorhandenen Beschreibungssätzen der von ihm eingetragenen Objekte/Module vorzunehmen. Die Ausgabe des jeweils angeforderten Beschreibungssatzes erfolgt in Form des unter *objekte.pl* dargestellten Formulars. Der Nutzer kann hier alle von ihm übermittelten Beschreibungsdaten korrigieren, wobei die bei der Übermittlung von Datenbeschreibungssätzen automatisch generierten Elemente nicht editierbar sind.

loeschen_objekte.pl: Dieses Programm bietet dem Nutzer die Möglichkeit, seine Objekte zu löschen. Der Nutzer erhält nach erfolgreichem Login eine tabellarisch geordnete Liste aller seiner Objekte und kann durch Markieren entscheiden welche Objekte er löschen möchte. Der anschließende Vorgang des Löschens ist irreversibel. In diesem Zusammenhang sei erwähnt, dass gelöschte Objekte durchaus Bestandteile von Lernobjekten (Lernmodulen, Lerneinheiten oder Lehrgängen) sein können. Lernobjekte werden durch diesen Vorgang nicht berührt, jedoch erhält derjenige, der die entsprechenden Lernobjekte zusammengestellt hat, eine e-Mail über die veränderte Ausgangssituation mit der Aufforderung, seine Lernobjekte zu modifizieren. Daher sollte der Vorgang des Löschens von Objekten durch den Eigentümer nur eine letzte, endgültige Möglichkeit darstellen: es ist besser, das schon eingetragene Objekt durch ein korrigiertes Objekt zu ersetzen!

lernmodul_loeschen.pl: Lernmodule stellen besonders einfach strukturierte Lernobjekte dar. Sie finden im Gegensatz zu Lerneinheiten keine weitere Verwendung im Sinne des modularen Ansatzes und lassen sich daher jederzeit auf Nutzer-Ebene durch den Eigentümer löschen.

Neben der Möglichkeit, Objekte durch den Eigentümer zu löschen, hat letztlich die User-Community das Recht, ein Objekt/Modul zu bewerten [Abbildung 15: Modulare Kopplung von Objekt und Objekt-Bewertung]. Sie entscheidet so über ein auf administrativer Ebene einzuleitendes Löschen oder gibt via e-Mail Hinweise zur Modifizierung an den Eigentümer weiter [Wiki, 2002].

Über alle Eintragungen von Objekten/Modulen wird sowohl der jeweilige Anwender als auch der Systemadministrator der *datenbankbasierten Entwicklungsumgebung* via e-Mail informiert. Jeder registrierte Nutzer hat die Möglichkeit, seine eingetragenen Objekte/Module über seinen nutzerspezifischen, passwortgeschützten Zugang zu administrieren. Eine SSL-Unterstützung für solche Operationen kann nach Anregung durch die User-Community umgesetzt werden.

4.2.4 Nutzerschnittstelle INTEGER

Einmal eingetragene Objekte lassen sich durch INTEGER (Integrierte Entwicklungsumgebung für eine Generierung von Lernmodulen, Lerneinheiten und Lehrgängen) zu modular aufgebauten Lernobjekten zusammenfügen, wobei unterschiedliche oder auch komplett neue Schwerpunktsetzungen möglich sind.

Der Aufbau von INTEGER und die damit verbundene Vorgehensweise orientiert sich am Vorhaben des jeweiligen Nutzers. Bevor exemplarisch die Generierung einer Lerneinheit beschrieben wird, soll vorab der grundsätzliche Aufbau von INTEGER erläutert werden: INTEGER teilt den **Hauptbildschirm** [Abbildung 20: Frames der *datenbankbasierten Entwicklungsumgebung*] in zwei skalierbare Frames.

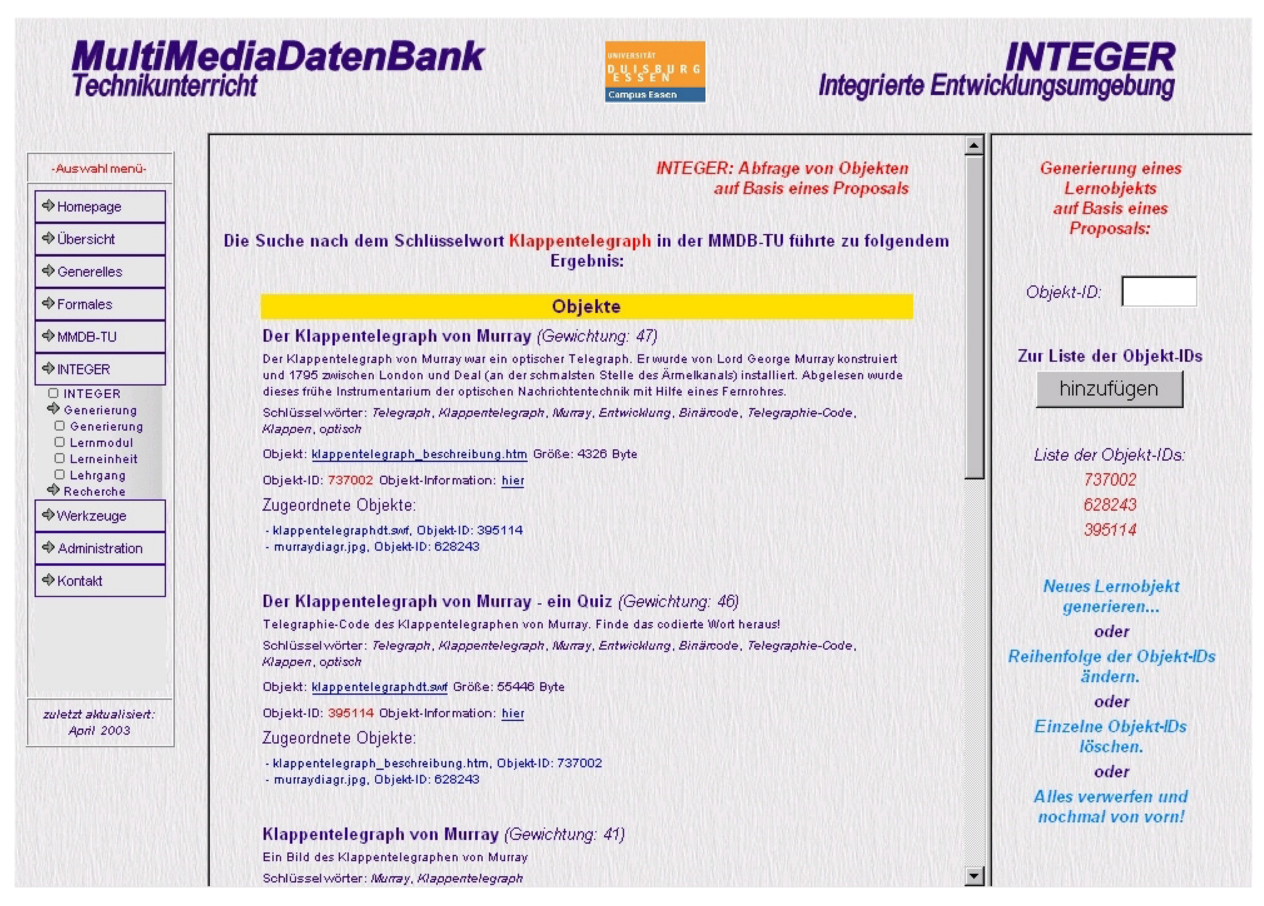


Abbildung 25: Frames der Nutzerschnittstelle INTEGER

Mit Hinblick auf das später dargestellte Beispiel der Generierung einer Lerneinheit ist in Abbildung 25 zur Verdeutlichung des Aufbaus eine solche Generierung auf Basis des Suchbegriffs **Klappentelegraph** wiedergegeben. Der linke Teil des Hauptbildschirms gibt das Ergebnis einer einfachen Suchabfrage nach einem Schlüsselwort innerhalb der Objekte der Datenbank *mmdb* wieder. Jedes gefundene Objekt wird in seinem Aufbau in identischer Form dargestellt. In diesem Fall sind das: Titel mit Gewichtung, Beschreibung, Schlüsselwörter, Objekt und Größe, Objekt-ID und Objekt-Information, Zugeordnete Objekte mit Dateinamen und zugehöriger Objekt-ID. Wie weiter oben erwähnt [Abbildung 11: Konzeption zur Generierung einer Lerneinheit] liefert die Abfrage von Objekten zum Zwecke der Generierung einer Lerneinheit[1] eine geordnete Liste[2], die zusätzlich thematisch verwandte Objekte als Empfehlungen enthält. Wichtig ist hier die Angabe der **Objekt-ID**. Sie kann in den rechten Frame des Hauptbildschirms in das dafür vorgesehene Feld eingetragen werden und anschließend einer Liste der Objekt-IDs hinzugefügt werden. Enthält diese Liste zwei oder mehr Objekt-IDs, so stehen dem

[1] Unter INTEGER wird eine Lerneinheit auch als Proposal bezeichnet. Für Lernmodule ist die Bezeichnung Sammlung und für Lehrgänge die Bezeichnung Variante gängig.

[2] Die Reihenfolge der aufgelisteten Objekte wird durch die Gewichtung der Objekte bezogen auf ein angefragtes Schlüsselwort wiedergegeben.

Nutzer vier Möglichkeiten einer weiteren Vorgehensweise offen: er kann ein *neues Lernobjekt (Lerneinheit) generieren*, er kann die *Reihenfolge der Objekt-IDs ändern*, er kann *einzelne Objekt-IDs löschen* oder a*lles verwerfen und noch einmal von vorne beginnen*. Im linken Frame des Hauptbildschirms bietet sich auch die Möglichkeit einer erneuten Abfrage mit einem anderen Schlüsselwort und einer veränderten Gewichtung.

Besondere Bedeutung hat der im rechten Frame des Hauptbildschirms dargestellte Punkt ***Neues Lernobjekt generieren...***: der Nutzer erhält hierüber den Zugang zu weiteren Optionen, wie *Generierung der durch die Objekt-IDs bestimmte Lerneinheit*, *Darstellung der künftigen Lerneinheit in der Bildschirmansicht* und die Möglichkeit, sie als *neue Lerneinheit* in die Datenbank *mmdb* einzutragen. Abbildung 26 zeigt das Struktogramm zur Generierung einer Lerneinheit mit den daran beteiligten Programmen. Der **passive Datenbank-Zugriff** wird im wesentlichen durch fünf PHP-Programme ausgeführt, der **aktive Datenbank-Zugriff**, nämlich die Eintragung der neuen Lerneinheit in die Datenbank *mmdb* allein durch das Perl-Programm *bm_parsing_proposal.pl*.

Um eine Lerneinheit zu erzeugen, ist über das PHP-Programm *login_proposal.php4* ein Login mit Nutzernamen und Passwort notwendig. Nach erfolgreicher Anmeldung übernimmt das PHP-Programm *integer_proposal.php4* die Frame-Steuerung des Hauptbildschirms: im linken Frame des Hauptbildschirms generiert das PHP-Programm *integer_proposal_abfrage.php4* dynamisch die gewichteten Ergebnisse einer Anfrage, während der rechte Frame die Möglichkeit zu einer Kombination von Objekten durch das PHP-Programm *integer_proposal_lernobjekt.php4* bietet. Wenn der modulbildende Vorgang durch den Nutzer abgeschlossen werden soll, wird über den Punkt ***Neues Lernobjekt generieren...*** das PHP-Programm *integer_proposal_lernobjekt_erzeugen.php4* aufgerufen. Dieses Programm generiert auf der Basis der zusammengestellten Objekt-IDs die neue Lerneinheit und stellt diese dem Nutzer auf Anforderung am Bildschirm zur Verfügung. Danach kann der Nutzer wählen, ob die neue Lerneinheit in die Datenbank *mmdb* aufgenommen werden soll oder nicht. Dieser Vorgang wird durch das Perl-Programm *bm_parsing_proposal.pl* abgeschlossen.

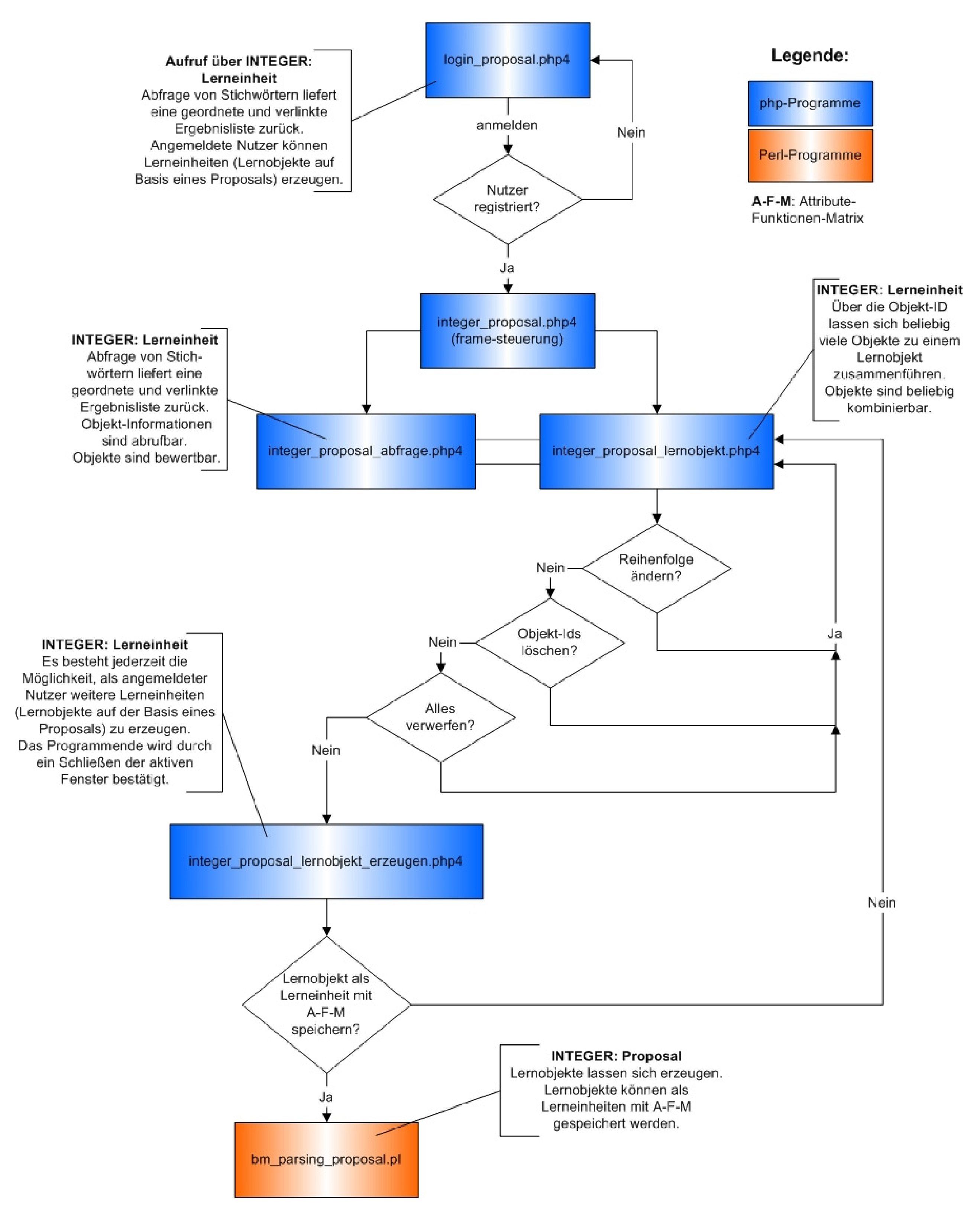


Abbildung 26: Generierung einer Lerneinheit (Struktogramm)

Wie schon im vorhergehenden Kapitel soll auch an dieser Stelle ein Programm durch die Darstellung von Quellcode-Sequenzen exemplarisch vorgestellt werden. Dazu wird

das Programm *integer_proposal_lernobjekt.php4* aus der vorhergehenden Abbildung 26 gewählt: es hat die Aufgabe, über eine extern angelegte Datei die Objekt-IDs aufzunehmen, die Reihenfolge dieser Objekt-IDs zu ändern, einzelne dieser Objekt-IDs zu löschen oder aber den modulbildenden Vorgang noch einmal neu zu starten. Des weiteren soll es dem PHP-Programm *integer_proposal_lernobjekt_erzeugen.php4* sämtliche Daten übergeben, um die gewünschte Lerneinheit zu generieren. Das Struktogramm dieses PHP-Programms soll an dieser Stelle durch einzelne, exemplarisch ausgewählte Code-Segmente beschrieben werden
[Abbildung 27: Programm-Flussbild integer_proposal_lernobjekt.php4].

Durch die Initialisierung erfolgt neben einer Einbindung des ausgelagerten PHP-Moduls `mmdb-tu.inc.php4` zur Herstellung einer Verbindung mit der Datenbank *mmdb*

```
<?php
     include("../php4/mmdb-tu.inc.php4");
     $connid = connect_to_mmdb();
?>
```

zusätzlich die Überprüfung der Datei *modul_proposal.txt* zur Aufnahme der Objekt-Liste, die bei Vorhandensein gelöscht wird. Die Angabe einer ungültigen Objekt-ID führt zu einer *Fehlermeldung*:

```
$result = mysql_query("SELECT objekt FROM objekte WHERE
                       objekt_id='$objekt_id'");
    if(mysql_num_rows($result)!=1)
    {
     echo "<p>";
     echo "<div align=\"center\"><font color=\"#FF0000\"><font
            face=\"arial\" size=\"2\"><b>Objekt-ID existiert
            nicht!</b></font></font></div>";
     MYSQL_CLOSE();
    }
```

Eine der wichtigsten Eigenschaften des Programms ist die beliebige Variation der Objekt-IDs, um eine nutzerspezifische Reihenfolge festzulegen. Dafür wurde die folgende Programmsequenz entwickelt, welche den Rang einer Objekt-ID mit deren Inhalt in Form einer Dezimalzahl verbindet. Erst dann besteht die Möglichkeit die so entstandenen Rang-Paare mit Hilfe des `usort`-Befehls nach den Vorgaben des jeweiligen Nutzers anzuordnen, wobei vorher eine Aufspaltung (`split`-Befehl) der Rang-Paare erfolgt:

```
for ($i=0; $i<$anzahl_objekt_id; $i++)
    {
     $rang_paare[]=$reihenfolge[$i].".".$array_objekt_ids[$i];
    }
function cmp ($a, $b)
         {
          if ($a == $b) return 0;
          return ($a < $b) ? -1 : 1;
         }
         $a = $rang_paare;
         usort ($a, "cmp");
    for ($i=0; $i<$anzahl_objekt_id; $i++)
        {
         list ($key, $value) = each ($a);
         $rang_paare_neu[] = $value;
         $aufspaltung = split("\.", $rang_paare_neu[$i]);
         $objekt_id_folge_neu[] = $aufspaltung[1];
        }
```

Auch der Vorgang des Löschens einzelner Objekt-IDs wird nach dem gleichen Verfahren der Zerlegung von Rang-Paaren durchgeführt:

```
for ($i=0; $i<$anzahl_objekt_id; $i++)
     {
      $del_paare[]=$objekt_del[$i]."."$array_objekt_ids[$i];
      $zerlegung = split("\.", $del_paare[$i]);
      if ($zerlegung[0] == "")
          {
           $objekt_id_folge_neu[] = $zerlegung[1];
          }
     }
```

Abschließend wird eine neue nutzerspezifische Datei für die Aufnahme der veränderten Objekt-IDs angelegt und die neue Reihenfolge der Objekt-IDs durch ein Array eingelesen:

```
$handle_modul_proposal = fopen("$objekt_id_proposal","a");
/* objekt-id in temporaere datei eintragen */
    for ($i=0; $i<$anzahl_objekt_id; $i++)
    {
    fputs($handle_modul_proposal, $objekt_id_folge_neu[$i]);
/* und den zeilenumbruch nicht vergessen */
/*   fputs($handle_modul_proposal, "\n"); */
   }
fclose($handle_modul_proposal);
```

Damit sind einige der Code-Segmente, aus denen sich das PHP-Programm aufbaut, dargestellt worden. Der komplette Quellcode kann, wie schon erwähnt, unter http://www.mmdb-tu.de/programmcode/quellcodes.htm abgerufen werden, da allein dieses PHP-Programm einen Umfang von 12 DIN A4 Seiten hat und daher an dieser Stelle ebenfalls nur auszugsweise dargestellt werden konnte.

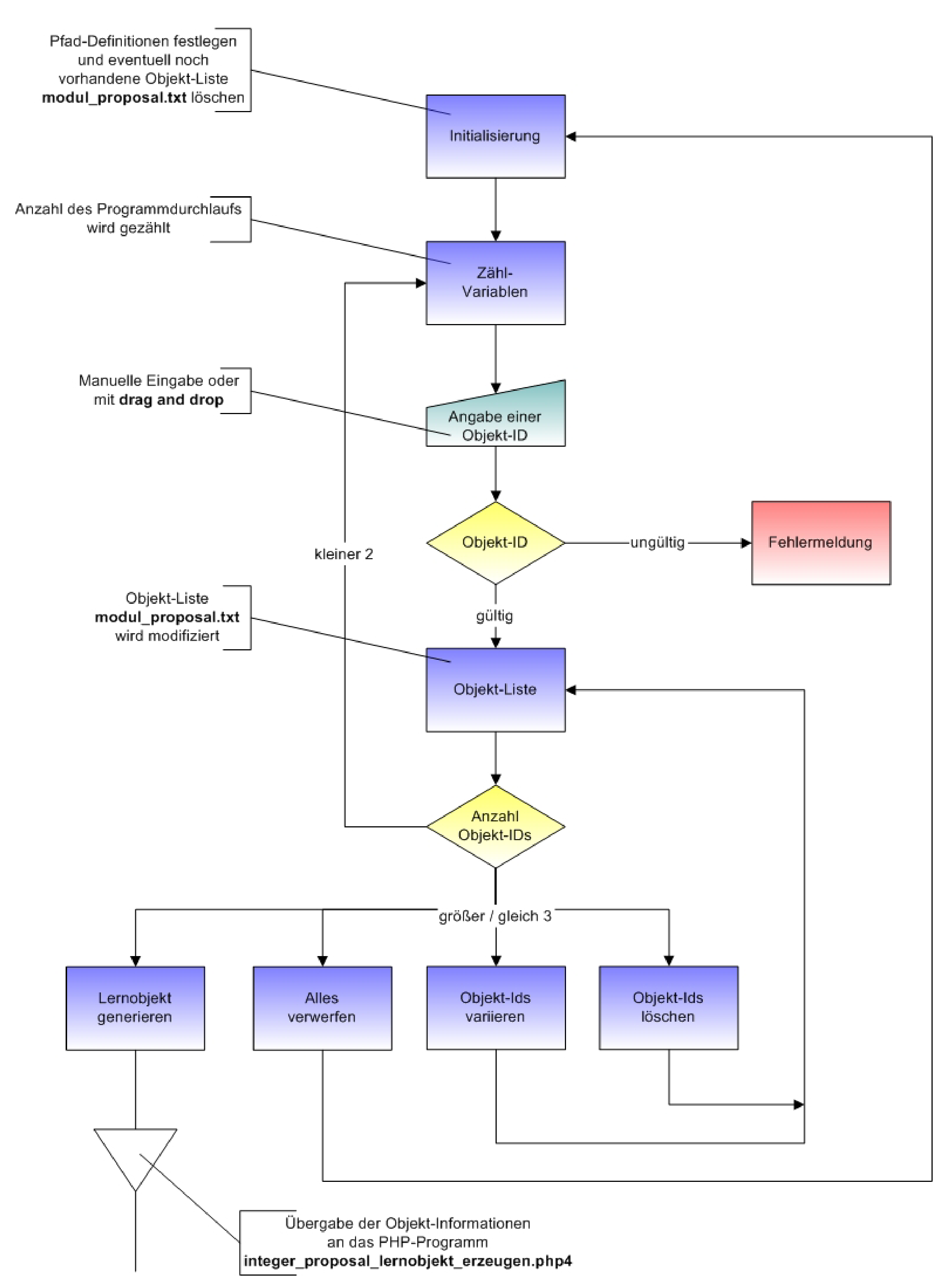


Abbildung 27: Programm-Flussbild *integer_proposal_lernobjekt.php4*

Unter der Nutzerschnittstelle INTEGER arbeitet eine Vielzahl von Programmen zusammen. Abbildung 28 stellt als Auszug der Sitemap der *datenbankbasierten Entwicklungsumgebung* dar, welche Programme hauptsächlich dazu beitragen, diese Nutzerschnittstelle zu realisieren. Es sind PHP-Programme, welche die passiven Datenbank-Zugriffe ausführen, während aktive Datenbank-Zugriffe, wie das Eintragen von neu generierten Lernmodulen, Lerneinheiten und Lehrgängen in die Datenbank *mmdb*, von Perl-Programmen übernommen wird.

Schon die Abbildung 26 hat deutlich gemacht, dass beispielsweise das PHP-Programm *integer_proposal.php4* lediglich das Steuerprogramm für 3 weitere PHP-Programme darstellt, die in der durch Abbildung 28 wiedergegebenen Sitemap der Nutzerschnittstelle INTEGER aus Gründen der Übersicht nicht aufgeführt sind[1]. Auch hier soll vor einer **Kurzbeschreibung der PHP-Programme** im einzelnen, zuerst der Aufbau dieser Sitemap erläutert werden:

Ausgehend von der root (http://www.mmdb-tu.de und http://www.integer-tu.de), die als durchgezogene Linie im äußerst linken Bildteil dargestellt ist, findet sich der Menüpunkt *INTEGER* (1. Spalte[2]). Von hier zweigen die beiden Untermenüs *Generierung* und *Recherche* ab (2. Spalte). Die 3., 4. und Spalte bietet nach einem Login die Möglichkeit, Lernobjekte (Lernmodule, Lerneinheiten und Lehrgänge) zusammenzustellen. Die von der 4. Spalte abzweigende Programmstruktur ist nicht mehr Teil das Java-basierten Auswahlmenüs der *datenbankbasierten Entwicklungsumgebung*, sondern basiert auf den für die Nutzer unsichtbaren Steuerdateien für die Generierung entsprechender Lernobjekte. Tabelle 9 zeigt bzgl. der jeweiligen Steuerdatei eine Übersicht der zugeordneten PHP-Programme.

Steuerdatei	zugeordnete PHP-Programme
integer_sammlung.php4	*integer_sammlung_abfrage.php4* *integer_sammlung_lernobjekt.php4* *integer_sammlung_lernobjekt_erzeugen.php4*
integer_proposal.php4	*integer_proposal_abfrage.php4* *integer_proposal_lernobjekt.php4* *integer_proposal_lernobjekt_erzeugen.php4*
integer_variante.php4	*integer_variante_abfrage.php4* *integer_variante_lernobjekt.php4* *integer_variante_lernobjekt_erzeugen.php4*

Tabelle 9: Steuerdateien mit zugeordneten PHP-Programmen

[1] Das Hilfesystem ist in Abbildung 28 ebenfalls nicht dargestellt.

[2] Die Anordnung der Spalten 1-5 verläuft von links nach rechts.

Nachdem durch die PHP-Programme

integer_sammlung_lernobjekt_erzeugen.php4,
integer_proposal_lernobjekt_erzeugen.php4 oder
integer_variante_lernobjekt_erzeugen.php4

entsprechende Lernobjekte erzeugt wurden, können diese neuen Lernobjekte durch die jeweils zugeordneten Perl-Programme

bm_parsing_sammlung.pl,
bm_parsing_proposal.pl oder
bm_parsing_variante.pl

in die Datenbank *mmdb* eingetragen werden (5. Spalte). Die 3. bzw. 5. Spalte bietet die Möglichkeit einer Bewertung der zur Verfügung stehenden Objekte, Lerneinheiten oder Lehrgänge.

In der 3. Spalte ermöglicht es der Menüpunkt *Recherche*, innerhalb der Datenbank *mmdb* direkt nach Objekten oder Lernmodulen zu suchen: diese beiden Bereiche sind ohne Login frei zugänglich. Allerdings lassen sich hier keine Objekte oder Lernmodule auf den eigenen Rechner herunterladen oder als Direktkopie vom Bildschirm speichern [Kapitel 4.2.6 Sicherheitsaspekte]. Einzig die Recherche von Lehrgängen erfordert ein Login. Eine Recherchemöglichkeit von Lerneinheiten ist hier nicht vorgesehen, da diese schon bei der Generierung von Lehrgängen implementiert ist [Abbildung 14: Modularer Aufbau passiver Datenbank-Zugriffe].

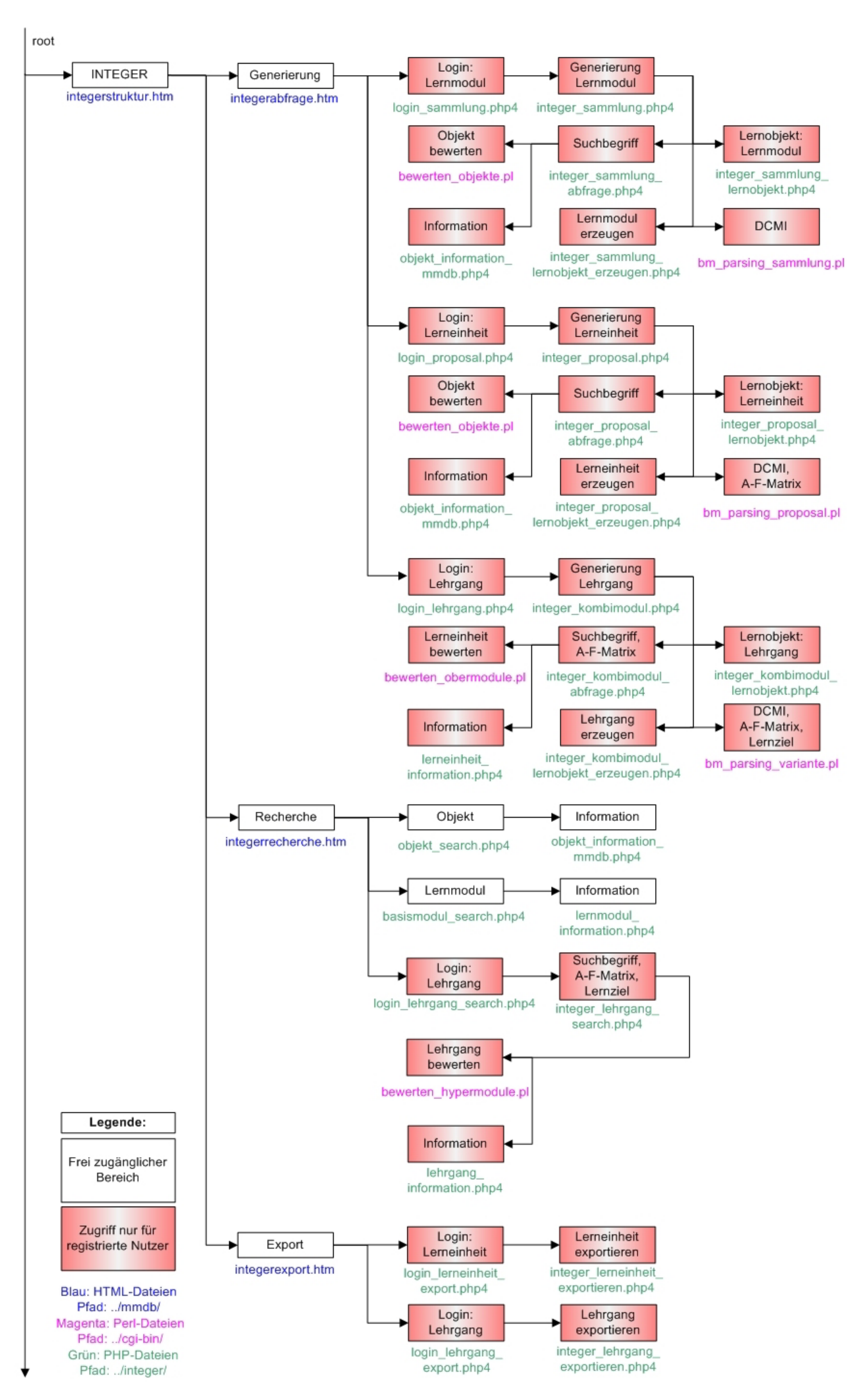


Abbildung 28: Sitemap der Nutzerschnittstelle INTEGER

Kurzbeschreibung der PHP-Programme

Sämtliche PHP-Programme mit den Namensteilen *_abfrage.php4*, *_lernobjekt.php4* und *_lernobjekt_erzeugen.php4* sind vom Grundprinzip her ähnlich: in Abhängigkeit davon, welche Art von Lernobjekt erzeugt werden soll, wird über die zugehörigen Steuerdateien aus Tabelle 9 die Generierung einer Sammlung (Lernmodul), eines Proposals (Lerneinheit) oder einer Variante (Lehrgang) realisiert. Die erfolgreiche Generierung wird über die jeweils zugeordneten Perl-Programme mit den Namensteilen *_sammlung.pl*, *_proposal.pl* und *_variante.pl* durch eine dynamisch erzeugte Webpage mit den entsprechenden Beschreibungsdaten sowie durch eine e-Mail mit allen zum Lernobjekt eingetragenen und automatisch generierten Angaben bestätigt.

Die in der linken Spalte von Tabelle 9 dargestellten PHP-Programme steuern die Generierung von Lernobjekten, wobei sie über die erste jeweils zugeordnete Datei der rechten Spalte aus Tabelle 9 im **linken Frame des Hauptbildschirms** [Abbildung 25: Frames der Nutzerschnittstelle INTEGER] ein Eingabefenster für Suchbegriffe aufbauen, um in der Datenbank *mmdb* Objekte bzw. Lerneinheiten, in deren Datenbeschreibungssätzen der gesuchte Begriff vorkommt, zu recherchieren. Anschließend werden die gefundenen Objekte bzw. Lerneinheiten in der Reihenfolge des Auffindens durch eine dynamisch erzeugte Webpage als Ergebnis-Liste ebenfalls im linken Frame des Hauptbildschirms wiedergegeben. Die Ergebnis-Liste enthält die folgenden Mindest-Angaben: Titel, Beschreibung, Schlüsselwörter, Objektname (als Hyperlink ausgeführt), Objektgröße (in Byte), Objekt-ID, Objekt-Information (als Hyperlink ausgeführt).

Der **rechte Frame des Hauptbildschirms** [Abbildung 25: Frames der Nutzerschnittstelle INTEGER] baut sich über die zweite jeweils zugeordnete Datei der rechten Spalte aus Tabelle 9 auf und bietet die Möglichkeit, Objekte bzw. Lerneinheiten in eine Objekt- bzw. Lerneinheit-Liste aufzunehmen. Diese Liste wird dann bei dem Vorgang der Generierung des entsprechenden Lernobjekts abgearbeitet, d. h. die einzelnen Objekte bzw. Lerneinheiten werden entsprechend der Reihenfolge der Objekt-IDs bzw. Lerneinheit-IDs in Form eines neuen Lernobjekts wiedergegeben.

Die Bildschirmdarstellung erfolgt über das jeweils letzte PHP-Programm der rechten Spalte aus Tabelle 9. Im folgenden werden daher nur die wesentlichen Unterschiede bei der **Generierung von Lernmodulen** [Abbildung 33: Screenshot: Neues Lernmodul] und bei der **Generierung von Lerneinheiten** [Abbildung 34: Screenshot: Neue Lerneinheit] erläutert[1].

Generierung von Lernmodulen

Die in Tabelle 9 wiedergegebenen Programme mit den Namensteilen *sammlung* steuern die Generierung von Lernmodulen. Für die Darstellung des linken Frames des Hauptbildschirms ist das PHP-Programm *integer_sammlung_abfrage.php4* zuständig, für die des rechten *integer_sammlung_lernobjekt.php4*. Das erste PHP-Programm liefert nach der Abfrage von Suchbegriffen eine ungeordnete Ergebnisliste verfügbarer Objekte [Abbildung 14: Modularer Aufbau passiver Datenbank-Zugriffe].

[1] Die Generierung von Lehrgängen verläuft im Prinzip so, wie die Generierung von Lerneinheiten, wobei aber nicht Objekte, sondern Lerneinheiten die Grundlage der Recherche bilden.

Mit Hilfe des zweiten PHP-Programms können ausgewählte Objekt-IDs einer Objekt-Liste, die nachträglich nicht mehr verändert werden kann, hinzugefügt werden. Diese Objekt-Liste gibt die Reihenfolge der Objekte bei der Generierung des Lernmoduls vor. Die Generierung selbst erfolgt durch das PHP-Programm *integer_sammlung_lernobjekt_erzeugen.php4*.

Die Eintragung des neu generierten Lernmoduls in die Datenbank *mmdb* (aktiver Datenbank-Zugriff) erfolgt durch das Perl-Programm *bm_parsing_sammlung.pl*: in diesem Fall muss ein neuer Datenbeschreibungssatz nach DCMI erstellt werden.

Die Generierung von Lernmodulen oder Sammlungen ist bewusst einfach gehalten: es wird hier die Möglichkeit geboten, einfach aufgebaute Module, die nicht unbedingt für den späteren Gebrauch weiter zur Verfügung stehen sollen, schnell zu erstellen. Die Ausgabe der Ergebnis-Liste baut sich schnell auf, da keine Sortieralgorithmen verlangsamend eingreifen. Neben dem Vorteil der Schnelligkeit weist diese Vorgehensweise den Nachteil einer ungenaueren Recherche auf.

Generierung von Lerneinheiten

Die in Tabelle 9 wiedergegebenen Programme mit den Namensteilen *proposal* steuern die Generierung von Lerneinheiten. Für die Darstellung des linken Frames des Hauptbildschirms ist das PHP-Programm *integer_proposal_abfrage.php4* zuständig, für die des rechten *integer_proposal_lernobjekt.php4*. Das erste PHP-Programm liefert nach der Abfrage von Suchbegriffen eine **geordnete** und **verlinkte** Ergebnisliste verfügbarer Objekte [Abbildung 14: Modularer Aufbau passiver Datenbank-Zugriffe und Abbildung 25: Frames der Nutzerschnittstelle INTEGER].

Geordnet heißt, dass die Ergebnis-Liste durch eine Gewichtung[1] der eingetragenen Suchbegriffe erzeugt wurde, wobei zwei einfache Regeln befolgt werden:

- Ein Treffer in der ersten Position erhält die höchste Gewichtung
- Taucht ein Treffer innerhalb der vorgegebenen Suchtiefe mehrfach auf, so wird seine niedrigste Gewichtung mit der Anzahl seines Auftretens multipliziert.

Jedes aufgelistete Objekt enthält daher neben seinen Mindest-Angaben zusätzlich die Angabe seiner auf die Suchbegriffe bezogenen Gewichtung. Diese Angabe ermöglicht bei einer erneuten Abfrage eine Reduzierung der Ergebnis-Liste zugunsten höherer Gewichtungen.

Verlinkt heißt, dass zusätzlich zu den Mindest-Angaben und der Angabe der Gewichtung eine Empfehlung zugeordneter Objekte erfolgt. Diese Objekte werden mit Objektnamen (als Hyperlink ausgeführt) und zugehöriger Objekt-ID angegeben.

Mit Hilfe des zweiten PHP-Programms können ausgewählte Objekt-IDs einer Objekt-Liste, deren Überarbeitung (die Reihenfolge der Objekt-IDs ist veränderbar, einzelne Objekt-IDs lassen sich löschen) jederzeit möglich ist, hinzugefügt werden. Diese Objekt-

[1] **Thema:** Der Suchbegriff wird unter den ersten 10 Wörtern gesucht. Gewichtung: **10** bis **1**
Beschreibung: Der Suchbegriff wird unter den ersten 20 Wörtern gesucht. Gewichtung: **20** bis **1**
Schlüsselwörter: Der Suchbegriff wird unter den ersten 20 Wörtern gesucht. Gewichtung: **20** bis **1**
Objekt-Name: Der Suchbegriff wird im Objekt-Namen gefunden. Gewichtung: **5**
Zugeordnete Links: Der Suchbegriff wird im ersten Link gefunden. Gewichtung: **5**
Die hier gewählten Zahlen geben ein willkürliches Maß wieder.

Liste gibt die Reihenfolge der Objekte bei der Generierung der Lerneinheit vor. Die Generierung selbst erfolgt durch das PHP-Programm *integer_proposal_lernobjekt_erzeugen.php4*.

Die Eintragung der neu generierten Lerneinheit in die Datenbank *mmdb* (aktiver Datenbank-Zugriff) erfolgt durch das Perl-Programm *bm_parsing_proposal.pl*: in diesem Fall muss ein neuer Datenbeschreibungssatz nach DCMI erstellt werden und es muss eine Eintragung in die A-F-Matrix erfolgen.

Die Generierung von Lerneinheiten ist ein Vorgang, der vom jeweiligen Nutzer mit Bedacht gewählt werden sollte. Die so erzeugten Lernobjekte stellen thematisch zusammenhängende Lerneinheiten dar, die für einen späteren Gebrauch bei der Generierung von Lehrgängen zur Verfügung stehen. Die Ausgabe der Ergebnis-Liste baut sich, bedingt durch die implementierten Algorithmen zur Gewichtung und Verlinkung langsamer auf, als bei der Generierung eines Lernmoduls, was aber bei Berücksichtigung der Ergebnisse nicht nachteilig ist.

Die **Generierung von Lehrgängen** erfolgt prinzipiell ähnlich, wie die gerade dargestellte Generierung von Lerneinheiten: die Ausgangsbasis ist in diesem Fall jedoch nicht das Objekt, sondern die Lerneinheit selbst. Lehrgänge werden auf der Grundlage von Lerneinheiten erzeugt, wofür letztlich das PHP-Programm *integer_variante_lernobjekt_erzeugen.php4* sorgt.

Die Eintragung des neu generierten Lehrgangs in die Datenbank *mmdb* (aktiver Datenbank-Zugriff) erfolgt durch das Perl-Programm *bm_parsing_variante.pl*: in diesem Fall muss ein neuer Datenbeschreibungssatz nach DCMI erstellt werden und eine Eintragung in die A-F-Matrix erfolgen; zusätzlich müssen Lernziele angegeben werden.

In jedem Fall wird sowohl der jeweilige Anwender als auch der Systemadministrator der *datenbankbasierten Entwicklungsumgebung* über alle Eintragungen von Lernobjekten wird via e-Mail informiert, wobei ebenfalls jeder registrierte Nutzer die Möglichkeit hat, seine eingetragenen Lernobjekte über seinen nutzerspezifischen, passwortgeschützten Zugang zu administrieren. Es bleibt anzumerken, dass sämtliche in die Datenbank *mmdb* eingetragenen Lernobjekte zusätzlich durch die User-Community bewertbar sind.

Als letzte Option von INTEGER ist der Export von Lernobjekten im XHTML-Format vorgesehen. Diese Option ist zeitlich unbestimmt, da in diesem Fall eine intensive Zusammenarbeit mit den Herstellern von Lern- und Kommunikationsplattformen notwendig ist. Dazu schreiben die Hersteller von ILIAS [ILIAS, 2003a]:

"Seit Beginn der Softwareentwicklung orientiert sich ILIAS an Standards im Bereich der Metadaten. Berücksichtigt wurden bislang IMS, ARIADNE und Dublin Core. Zur Zeit passen wir das ILIAS-Metadatensystem an die aktuelle Entwicklung an."

4.2.5 Administrativer Bereich

Dieser passwortgeschützte Bereich beinhaltet im wesentlichen Perl-Programme, die ein einfaches Administrieren der Datenbank *mmdb* ermöglichen.

Abbildung 24 gibt den in die Sitemap integrierten administrativen Bereich wieder, der hauptsächlich durch die Perl-Programme *zip_parsing.pl*, *lerneinheit_loeschen.pl* und *lehrgang_loeschen.pl* bestimmt ist.

Das Perl-Programm *zip_parsing.pl* wurde als Hilfsmittel erstellt, um die Datenbank *mmdb* schnell durch neue Datensätze zu ergänzen. Es erfordert zwei Dateien für jedes einzutragende Objekt. Alle Dateien sind jedoch zu einer einzigen Archiv-Datei *(*.zip)* zusammengefasst. Die Größe dieser Datei, sowie die Anzahl der in ihr enthaltenen Objekte kann dabei beliebig sein. Unterstützt werden dabei alle durch Tabelle 3 wiedergegebenen Medientypen[1]. Auf diese Art und Weise können sehr viele Objekte sehr schnell in die Datenbank *mmdb* eingetragen werden. Der komplette Quellcode kann unter http://www.mmdb-tu.de/programmcode/quellcodes.htm abgerufen werden; das Programm hat einen Umfang von ca. 30 DIN A4 Seiten.

Zusätzlich existieren die Perl-Programme *lerneinheit_loeschen.pl* und *lehrgang_loeschen.pl*, wobei schon deren Namen ihre jeweilige Funktion erläutern. Sie sind im Prinzip ähnlich aufgebaut, wie das Perl-Programm *loeschen_objekte.pl*. Auch in diesem Fall kann der komplette Quellcode unter
http://www.mmdb-tu.de/programmcode/quellcodes.htm abgerufen werden.

Die Administration der Datenbank *mmdb* auf Systemebene erfolgt durch das als PHP-Programm ausführbare graphische Frontend phpMyAdmin aus dem Bereich der Open-Source-Software [Abbildung 35: Screenshot: Administrativer Bereich, phpMyAdmin].

Weitere administrative Aspekte, insbesondere die Systemsicherheit betreffend, sind im nachfolgenden Kapitel über Sicherheitsaspekte dargelegt.

4.2.6 Sicherheitsaspekte

Jedes netzgebundene System kann so gut geschützt sein, wie es geht – es bleibt dennoch angreifbar. Das gilt nicht nur für die Client-Seite [Schmidt, 2002], sondern auch für die Server-Seite [Köster, 2002]. Eine weltweit über das Internet erreichbare Präsenz, in diesem Fall die *datenbankbasierte Entwicklungsumgebung*, muss über verschiedene Sicherheitsmechanismen verfügen, die in den Bereichen **Betriebssystem**, **Webserver**, **Datenbankserver** und **Nutzerschnittstellen** wirksam sind. Die im folgenden geschilderten Sicherheitsmechanismen stellen die wesentlichen Möglichkeiten dar, um diese Internet-Präsenz relativ sicher zu etablieren.

Betriebssystem

Für interne Rechner-Ports stehen 16 Adressleitungen zur Verfügung, d. h. es existieren 65536 verschiedene Ports: Privilegierte Ports (0-1023), Registrierte Ports (1024-49151) und Private Ports (49152-65535). Dabei sind lediglich die ersten 2^{10} Ports für die wichtigsten Dienste festgelegt [IANA, 2003]. Nach Einrichten des Betriebssystems wurden bis auf die Ports 21 (**ftpd**[2]), 22 (**ssh**[3]), 80 (**httpd**[4]) und 3306 (**mysqld**[5]) aus Sicherheitsgründen sämtliche weiteren Ports gar nicht erst geöffnet bzw. durch die Port-Filterung des HRZ gesperrt, weswegen auch auf den Einsatz einer Firewall verzichtet wurde. Ei-

[1] Im Anschluss an Tabelle 3 wird auf Einschränkungen hingewiesen.

[2] ftpd: file transfer protocol daemon

[3] ssh: secure shell

[4] httpd: hypertext transfer protocol daemon

[5] mysqld: mysql daemon

ne regelmäßige Sicherung über ein Bandserversystem ist erst geplant, wenn es der Umfang der Datenbank *mmdb* erfordert. Das wird erst dann der Fall sein, wenn die o.g. Internet-Präsenz uneingeschränkt über das WWW zur Verfügung steht.

Webserver

Die Datei **httpd.conf** zur Konfiguration des Apache Webservers wurde so eingerichtet, dass über PHP und Perl eine Datenbank auf MySQL-Basis zu Verfügung steht, die über den einzigen öffentlichen Nutzerzugang **~mmdb-tu** angesprochen werden kann [Wehling, 2003a]. In der Entwicklungsphase wurde ein Zugangsschutz für das Verzeichnis **public_html** durch die Datei **.htaccess** auf der Basis des Apache-Moduls *mod_auth*, sowie eine Sperre für Suchmaschinen durch die Datei **robots.txt** eingerichtet; der Zugangsschutz wurde später durch eine nutzerspezifische Abfrage ergänzt. So wurde zum einen sichergestellt, dass nicht laufend Suchroboter und Spider[1] das zu entwickelnde System zum Zwecke einer Indexierung störten, zum anderen sollten ausschließlich in der Datenbank registrierte User diese auch nutzen können. Außerdem wurden administrative Zugänge über **ssh** und **ftp** eingerichtet sowie alle weiteren notwendigen Dämonen über den Dämon **inetd**[2] eingebunden.

Durch regelmäßige Analyse der Access-Files mit Hilfe der Open-Source-Software AWStats [AWStats, 2003] wurden und werden immer wieder Wurmangriffe auf den Webserver festgestellt. Es handelt sich hierbei um Angriffe des CR I[3], erkennbar durch den folgenden Auszug einer Access-Datei:

```
GET /default.ida?NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
NNNNNNNNNN%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%
u7801%u9090%u9090%u8190%u00c3%u0003%u8b00%u531b%u53ff%u0078%u0000%u00=a
```

Angriffe dieser Art werden jedoch problemlos durch die beschriebene Konfiguration von Betriebssystemdiensten und Webserver abgewehrt [Unix, 1999].

Datenbank

Die Datenbank *mmdb* wurde mit einem Passwortschutz versehen, der ausschließlich einen Zugriff über Perl- und PHP-Programme gestattet. Die Nutzer-Authentifizierung erfolgt über die Datenbank *mmdb* selbst. Die Rechte dieser Nutzer richten sich nach den von Ihnen durchzuführenden Vorhaben; daher ist Vergabe von Nutzerrechten über den GRANT-Befehl[4] nicht vorgesehen.

Nutzerschnittstellen

Über Perl und PHP werden den Nutzern etliche Formulare zur Eintragung und Bearbeitung von Objekten/Modulen der *datenbankbasierten Entwicklungsumgebung* zur Verfügung gestellt. Jeder dieser Vorgänge erfordert ein Login des entsprechenden Nutzers

---

[1] Suchroboter und Spider sind webgebundene, automatisierte Suchprogramme, die auf Webseiten zum Zwecke einer Indexierung bei den jeweiligen Suchmaschinen zugreifen.

[2] inetd: internet daemon

[3] CR I steht für den Webwurm Code Red I, sein Angriff ist durch eine Häufung von 'N' nachweisbar. Der Nachfolger CR II macht sich stattdessen durch 'X' bemerkbar.

[4] Mit dem GRANT-Befehl werden Nutzer-Rechte vergeben. Leider findet man unter MySQL, je nach Version, recht eigenwillige Implementationen dieses Befehls. Vor MySQL 3.22.11 bewirkte der GRANT-Befehl gar nichts.

mit Nutzernamen und zugehörigem Passwort. Erst dann wird der Nutzer in der Lage sein, in Kontakt mit der Datenbank *mmdb* zu treten. Um fehlerhafte oder unzulässige Formular-Einträge zu verhindern, wurden sämtliche Formular-Felder mit speziellen Fehlerabfangroutinen versehen. Dazu sollen an dieser Stelle lediglich zwei kleine Beispiele genannt werden, die im wesentlichen auf eine Anwendung Regulärer Ausdrücke zurückgehen.

- Gültigen Nutzernamen überprüfen:
  ```
  /^[A-ZÄÖÜ]+[a-zäöü]+$/
  ```
- Gültigen URI überprüfen:
  ```
  /^http://(([a-zA-Z0-9]{1,3}[\.][a-zA-Z0-9]{1,}[\.][a-zA-
  Z]{1,3}$)|(localhost)|([a-zA-Z0-9]{1,3}[\.][a-zA-Z0-9]{1,}[\.][a-zA-
  Z]{1,3}[/].+$))/
  ```

Des weiteren können durch die Perl-Programme keine Shell-Commands ausgeführt werden, die Schaden anrichten könnten, wie z. B.:

```
rm -rf
```

Zusätzlich sind Filterfunktion eingesetzt worden, um beispielsweise die Größe der in die Datenbank *mmdb* übertragenen Objekte auf 1 MB zu beschränken[1].

Das Auswahlmenü der *datenbankbasierten Entwicklungsumgebung* ist in DHTML ausgeführt, d. h.: Java-Applets werden durch JavaScript-Code gesteuert. Das wiederum hat zur Folge, dass der Nutzer bei nicht aktiviertem JavaScript kein Auswahlmenü zur Verfügung hat! Er ist gezwungen die Scripting-Funktionen seines Browser freizuschalten. Sämtliche Pfad-Angaben, die bei einem Arbeiten mit der *datenbankbasierten Entwicklungsumgebung* auftauchen könnten, werden durch integrierte Javascript-Elemente unterdrückt. So öffnet die folgende Funktion

```
function fenster_objekt_information(objekt_information)
{
 fenster_objekt_information_oeffnen=window.open(objekt_information,
 '',toolbar=yes,scrollbars=yes,status=no,resizable=yes,width=500,
 height=800,screenX=100,screenY=100');
}
```

beispielsweise ein Fenster zur Darstellung der Objekt-Informationen, es wird jedoch durch die fehlende Statuszeile keine Pfadinformation über die Herkunft des betreffenden Objekts geliefert. Der Anspruch, Lernobjekte ausschließlich durch INTEGER zu generieren wird zusätzlich dadurch unterstrichen, dass sämtliche generierten Lernobjekte, die in der Bildschirmansicht darstellbar sind, mit einem Java-Code versehen wurden, der dem Rechtsklick mit der Maus unterbindet, d. h.: weder Objekte noch Lernmodule lassen sich durch Rechtsklick speichern. Wenn Lernobjekte zwecks Weiterverarbeitung zur Verfügung stehen sollen, muss die Export-Funktion von INTEGER genutzt werden.

---

[1] Diese Beschränkung ist notwendig, da nicht jeder Nutzer verfügt über einen breitbandigen Internetzugang verfügt.

5 Zusammenfassung und Ausblick

Die Durchführung von Projekten stärkt die Position des Faches Technik sowohl auf Länder- als auch auf Bundesebene. Der Bogen lässt sich hier von länderspezifischen Projekten über länderverbindende Projekte [Kapitel 2.3 Fachspezifisch] bis hin zu einer Projekt-Beteiligung auf EU-Ebene spannen. Für das Fach Technik wurden Projekte gewählt, die einen informationstechnischen Schwerpunkt aufwiesen, der systembedingt Elemente aus den angrenzenden Bereichen des Stoff- und Energieumsatzes sowie der Soziotechnik einschloss. Im Bereich des dadurch akzentuierten e-Learnings sprechen die vom Fach durchgeführten Aktivitäten [Kapitel 3.1.4 e-Learning / Blended Learning] der letzten drei Jahre für sich und weisen als die künftige Richtung eine europa- und schließlich weltweite Zusammenarbeit aus.

Die Stellung des Faches Technik im Umfeld von Schule und Hochschule wird z. Zt. durch die Fusion der beiden Universitäten Duisburg und Essen zur Universität Duisburg-Essen mit jeweils spezifizierten Campi forciert. Im Rahmen der Fusionsgespräche wird wahrscheinlich das Fach Technik gestärkt hervorgehen, da der Campus Essen u. a. zu einem Zentrum der Lehrerbildung ausgebaut werden soll. Das Fach Technik am Campus Essen würde dabei durch die langjährige Erfahrung bei der Durchführung von Projekten, insbesondere mit informationstechnischem und multimedialem Schwerpunkt, eine sehr gute Ausgangsbasis bieten und so die reibungslose Kooperation zwischen beiden Universitäten im Rahmen einer Qualitätsverbesserung des Lehrangebots sowohl in Duisburg als auch in Essen ermöglichen.

In diesem Sinne wird mit der Durchführung von entsprechend angelegten Projekten genau das erreicht, was die Position des Faches Technik fordert [Kapitel 3.2 Forderungen]: die Verbesserung der Studierbarkeit des Faches, die Erweiterung der medialen Kompetenzen, der adäquate Einsatz der Neuen Medien und der über allem stehende und damit unlösbar verbundene didaktische Mehrwert [Kapitel 2.3 Fachspezifisch].

Das wesentliche Ergebnis der vorliegenden Arbeit stellt die Schaffung der *datenbankbasierten Entwicklungsumgebung* selbst dar. Dieses mittlerweile im Seminareinsatz unverzichtbare Instrumentarium wird von den Nutzern als das angenommen, wofür es gedacht und entwickelt worden ist: als Möglichkeit, die Lücke zwischen unstrukturiertem Content und effektivem e-Learning zu schließen, wobei das dem modularen Ansatz zugrunde liegende dreistufige Modell eines Lernens mit multimedialen Inhalten die fachdidaktische Basis bildet. Damit stellt die *datenbankbasierte Entwicklungsumgebung* in der Tat eine grundsätzliche Lösung der am Anfang genannten Kernfrage dar [Kapitel 3 Problemstellung]. Zu einer abschließenden Beantwortung dieser Kernfrage muss eine Bestandaufnahme in Form einer expliziten Betrachtung der Umsetzungskriterien [Kapitel 3.2.2 Realisierungsmöglichkeit] und der daraus resultierenden Umsetzungsergebnisse [Kapitel 3.2.3 Zielsetzung] erfolgen. Dabei ergibt sich das folgende Bild:

Die geforderte **Browserunabhängigkeit** lässt sich nur bedingt verwirklichen, wird in absehbarer Zeit aber auch kein ernst zu nehmendes Problem mehr darstellen [Abbildung 19: Analyse der Browser-Anteile [Fittkau, 2003]].

Ein **ausgewogenes Webdesign** wurde durch eine nutzerfreundliche Aufbereitung von relevanten Inhalten realisiert. Es wurde darauf geachtet, dass eine ausgewogene Nutzerführung die durch die Website dargestellten Inhalte in sinnvoll gegliederter Weise anwählbar macht. Die gewählte Frame-Darstellung mit integrierten Elementen der

Fenstertechnik entspricht mittlerweile einem vertrauten Webdesign und hilft dem Nutzer, sich schnell in der *datenbankbasierten Entwicklungsumgebung* zurecht zu finden. Unterstützt werden Aspekte der Navigation zusätzlich durch eine animierte, methodische System-Struktur, die alternativ genutzt werden kann [Abbildung 29: Struktur der *datenbankbasierten Entwicklungsumgebung*].

Mit Hinblick auf einen angestrebten europaweiten Einsatz liegt die *datenbankbasierte Entwicklungsumgebung* zusätzlich in englischer Sprache vor [siehe http://www.mmdb-tu.de und http://www.integer-tu.de]. Diese **Bilingualität** stellt für Lern- und Kommunikationsplattformen ein Ausschlusskriterium dar und muss von einem System, das als ein o.g. Bindeglied fungieren soll, erfüllt werden [Abbildung 36: Generation of a new Learning Unit].

Mit **Vermeidung nicht etablierter Standards** ist in diesem Fall die Ausführung sämtlicher Aktivitäten in HTML bzw. XHTML gemeint. Standards wie XML werden vorläufig nicht im Rahmen der *datenbankbasierten Entwicklungsumgebung* verfügbar sein. Für den Export vorbereitende, XML-basierte Lernobjekte werden erst dann verfügbar sein, wenn die Hersteller von Lern- und Kommunikationsplattformen diesen Ansatz einfordern. Das ist bis zum jetzigen Zeitpunkt jedoch nicht der Fall.

Die **Vermeidung proprietärer Standards** wurde durch den Ansatz nach L.A.M.P. weitgehend erfüllt, wobei zusätzlich, soweit möglich, auch Aspekte der Interoperabilität und Portabilität berücksichtigt wurden. Es bleibt zu erwähnen, dass RDBMS wie MySQL proprietäre Sprachelemente auf der Basis von SQL aufweisen. Solche Elemente wirken sich jedoch erst bei extrem komplexen Systemen mit hoher User-Frequentierung hemmend aus.

Sämtliche Objekte, die durch die Datenbank *mmdb* verwaltet werden, sind im wesentlichen auf eine Größe von 1 MB beschränkt. Diese Restriktion wurde gewählt, um **ladezeitoptimierte Inhalte** zu bevorzugen. Es kann davon ausgegangen werden, dass der durchschnittliche Nutzer nur über ein analog arbeitendes Modem verfügt. Bei dem höchst möglichen Datendurchsatz würde die Übertragung einer solchen Datei ca. 2 Minuten betragen, was gerade noch vertretbar erscheint. Auch die Transformierung der Datenbank *mmdb* in die 3. Normalform wird sich bei wachsender User-Frequentierung positiv auf die Recherche- und Ladezeiten auswirken. Zusätzlich wurde in der graphischen Umsetzung der Nutzerschnittstellen MMDB-TU und INTEGER darauf geachtet, dass der Seitenaufbau nicht durch aufwändige Graphiken unnötig verzögert wird. Innerhalb der Nutzerschnittstellen existieren daher lediglich drei Graphiken: die beiden Logos für MMDB-TU und INTEGER sowie ein e-Mail-Symbol.

Der **Einsatz interaktiver Elemente** wurde durch die Integration verschiedener Java-Applets vorgenommen. So ist beispielsweise die alternative Navigation über den Menüpunkt *Methodische Struktur* mit einer integrierten Suchfunktion versehen, welche durch Interaktion mit dem Nutzer die Navigation erheblich erleichtert. Weitere Elemente sind durch den Einsatz von JavaScript realisiert worden: die DC-Generatoren, die Template-Generatoren und die **integrierte Hilfefunktion** [Abbildung 32: Screenshot: Integrierte Hilfefunktion].

Einfache e-Mail-Dienste sind in sämtliche Perl-Programme integriert, um den Nutzer über seine jeweils durchgeführten Aktionen in dieser Form zu unterrichten [Kapitel 4.2.3 Nutzerschnittstelle MMDB-TU].

Für eine **schnelle und zuverlässige Internetanbindung** sollte das HRZ sorgen, während die **Ausfallsicherheit durch Redundanzhardware** im wesentlichen durch ein RAID-System realisiert ist. Spannungsschwankungen werden durch den Einsatz einer USV kompensiert.

Die gewählte Vorgehensweise, einen Ansatz nach L.A.M.P. zu verfolgen, hat sich in jeder Hinsicht bewährt: sämtliche dafür notwendige Software arbeitet ausgesprochen stabil und liefert auf der Basis der letzten Versionen ein sicherheitstechnisch unbedenkliches System, wenn bestimmte Vorkehrungen getroffen werden [Kapitel 4.2.6 Sicherheitsaspekte]. Ein wichtiges Instrumentarium der *datenbankbasierten Entwicklungsumgebung* ist ihre Pflege und Administration durch die User-Community. Diesem Umstand wird dadurch Rechnung getragen, dass sämtliche Objekte/Module bewertbar sind [Kapitel 4.1.3 Integration]. Auf der Grundlage dieser Bewertung kann die User-Community und/oder der Administrator entscheiden, welche Wartungsvorgänge durchzuführen sind und so gleichzeitig für die geforderte Nachhaltigkeit sorgen. Ähnliche, quasi sich selbst administrierende Systeme existieren schon und haben sich im alltäglichen Einsatz durchaus bewährt [Wiki, 2002]. Die Realisierung der *datenbankbasierten Entwicklungsumgebung* war sehr zeitintensiv und aufwändig, da neben Kenntnissen auf Unix-Ebene zusätzlich solche auf den folgenden wesentlichen Gebieten notwendig waren: Datenbankserver und -design, Webserver, HTML, XHTML, XML, RDF, PHP, Perl, Java, JavaScript, Webdesign, Hardware-Architekturen, Software-Produkte.

Die *datenbankbasierte Entwicklungsumgebung* stellt zwei Nutzerschnittstellen (MMDB-TU und INTEGER) zur Verfügung, die so aufgebaut sind, dass dem Nutzer keine tiefgreifenden informationstechnischen Kenntnisse abverlangt werden, um zu schnellen Ergebnissen zu gelangen. Eine zusätzliche Hilfe bietet das videobasierte Benutzer-Handbuch, das jederzeit online unter den beiden URI:

http://www.mmdb-tu.de/benutzerhandbuch/mmdb-tu_videos.htm bzw.
http://www.mmdb-tu.de/benutzerhandbuch/integer-tu_videos.htm

abrufbar ist.

Bezogen auf die Kernfrage lässt sich festhalten, dass die hier begründet gewählte Vorgehensweise sicherlich keine Ultima Ratio darstellt: sie ist ein möglicher Weg, auf dem ein multimedial orientiertes Lernen unterstützt werden kann. Es hat sich herausgestellt, dass dieser Weg gangbar ist und zum beabsichtigten Ziel führt, wobei angemerkt werden sollte, dass er zweifellos durch die *datenbankbasierte Entwicklungsumgebung* realisiert worden ist und nicht, wie viele andere Vorhaben ähnlicher Art, ein vorzeitiges Ende in einem dubiosen Planungsszenario fand.

Es bleibt zu erwähnen, dass z. Zt. über eine Trennung von Datenbankserver und Webserver, Replikationssysteme, Clusterbildung und die Implementierung eines Load Balancing nachgedacht wird. Ebenso, wie die Hardwareseite einer ständigen Überprüfung im Sinne einer realisierbaren Optimierung bedarf, wird die Softwareseite ebenfalls im Hinblick auf eine Verbesserung von Programmen geprüft. Es ist zu berücksichtigen, dass, bedingt durch laufende Modifikationen am Quellcode der hier beschriebenen Programme, die schriftlich vorliegende Form dieser Arbeit, bezogen auf den Quellcode, nur eine Momentaufnahme sein kann. Die aktuellen Änderungen finden sich im Internet unter dem bekannten URI

http://www.mmdb-tu.de/programmcode/quellcodes.htm.

Die Kernaussage dieser Arbeit bleibt dabei jedoch unangetastet.

Durch ihren prinzipiellen Aufbau und die Realisierung der vorgenannten Umsetzungskriterien weist *datenbankbasierten Entwicklungsumgebung* Ansatzpunkte für eine mögliche Vorgehensweise bei der Durchführung von EU-Projekten auf. So würde beispielsweise der Einsatz objektorientierter Datenbank-Module dem Nutzer die Möglichkeit bieten, seine eigene, auf seine ganz speziellen Bedürfnisse zugeschnittene Datenbank zu erzeugen. Die damit verbundenen Nutzerschnittstellen in Form von graphischen Frontends könnten dabei ebenfalls nutzerspezifisch zugeschnitten sein. Die grundlegende Software, um eine solches Vorhaben zu initiieren, wird nicht lange auf sich warten lassen, vorausgesetzt, die Community der Open-Source-Entwickler arbeitet weiterhin so effektiv wie bisher.

6 Anhang

6.1 Benutzer-Handbuch

Wie in der Kurzfassung sowie in der Zusammenfassung dieser Arbeit erwähnt, wurden an Stelle eines Benutzer-Handbuchs Video-Sequenzen als Dokumentationsform gewählt. Diese stehen unter

http://www.mmdb-tu.de/benutzerhandbuch/mmdb-tu_videos.htm bzw.
http://www.mmdb-tu.de/benutzerhandbuch/integer-tu_videos.htm

zum Download bereit[1].

6.2 Begriffsdefinitionen und -analogien

An dieser Stelle ist eine zusammenfassende Übersicht der Begriffsdefinitionen und –analogien wiedergegeben [Kapitel 3.2.1 Ansatzpunkt].

In der Anfangsphase der *datenbankbasierten Entwicklungsumgebung* wurden, um den modularen Charakter zu unterstreichen, Lernmodule Basismodule oder Module der Basisklasse genannt; Lerneinheiten wurden als Obermodule oder Module der Oberklasse bezeichnet und Lehrgänge als Hypermodule oder Module der Hyperklasse. Teilweise finden sich diese Bezeichnungen noch in den Programmen wieder.

Begriffsdefinitionen:

Objekt: Ein Objekt ist eine kleinste, nicht mehr weiter unterteilbare, zusammenhängende Einheit (z. B. ein Bild, ein thematisch geschlossener Text, ein Applet, eine Animation, etc.).

Modul: Ein einfaches Modul besteht aus wenigstens zwei Objekten, die zu einer thematisch zusammenhängenden Einheit gefügt sind. Ein Modul höherer Ordnung besteht aus wenigstens zwei einfachen Modulen, die zu einer komplexen thematisch zusammenhängenden Einheit gefügt sind (z. B. Lernmodul, Lerneinheit, Lehrgang).

Begriffsanalogien:

Lernobjekt: Bezeichnung für alle aus wenigstens zwei Objekten bestehenden Module. Damit sind Lernmodule, Lerneinheiten und Lehrgänge gemeint. Die folgende Tabelle 10 zeigt die innerhalb der Nutzerschnittstellen MMDB-TU und INTEGER verwendeten Begriffsanalogien.

1 Hier finden sich auch Beispiele für einen DC-konformen Aufbau von Objekten und Modulen. In diesem Zusammenhang ist die Fertigung einer menügeführten DVD geplant, da nicht jeder Nutzer über einen breitbandigen Internet-Zugang verfügt.

	MMDB-TU	INTEGER
Lernmodul	Modul der Basisklasse, Basismodul	Sammlung, Lernobjekt auf Basis einer Sammlung
Lerneinheit	Modul der Oberklasse, Obermodul	Proposal, Lernobjekt auf Basis eines Proposals
Lehrgang	Modul der Hyperklasse, Hypermodul	Variante, Lernobjekt auf Basis einer Kombination von Lerneinheiten

Tabelle 10: Begriffsanalogien MMDB-TU und INTEGER

6.3 Spezielle Beispiele

Die **Eintragung einer Lerneinheit** und die **Eintragung eines Lehrgangs** sollen in exemplarischer Form den Umgang mit der *datenbankbasierten Entwicklungsumgebung* veranschaulichen. An dieser Stelle finden sich keine Visualisierungen, sondern es wird lediglich gezeigt, welche Systematik den Eintragungen zugrunde liegt.

Eintragung eines Lerneinheit

Es soll eine Lerneinheit mit Hilfe des dafür zuständigen Perl-Programms *bm_parsing_proposal.pl* in die Datenbank *mmdb* eingetragen werden. Die Lerneinheit wurde vorher über die Nutzerschnittstelle INTEGER durch die Kombination von zwei Objekten zusammengestellt. Grundsätzlich ist ein Beschreibungssatz nach DC erforderlich, der an dieser Stelle jedoch nicht wiedergegeben wird, da die folgenden Objekte nur vage thematisiert werden.

Das erste Objekt ist ein einführender Text, der einen Stufenindex-Lichtwellenleiter bzgl. seiner numerischen Apertur beschreibt, das zweite Objekt ist eine Zeichnung, die den Strahlengang am Einspeisungsort des Lichtstrahls darstellt. Die Eintragung in die **A-F-Matrix** sieht folgendermaßen aus: Es wird das Attribut *Informationsumsatz* mit der Funktion *Transport* gewählt. Damit wird diese Lerneinheit dem Anwendungsfeld *Information und Kommunikation* zugeordnet. Das technische Sachsystem wird spezifiziert durch die Eintragung *Informationsübertragungstechnik.*

Eintragung eines Lehrgangs

Es soll ein Lehrgang mit Hilfe des dafür zuständigen Perl-Programms *bm_parsing_variante.pl* in die Datenbank *mmdb* eingetragen werden. Der Lehrgang wurde vorher über die Nutzerschnittstelle INTEGER durch die Kombination von zwei Lerneinheiten zusammengestellt. Auch hier ist ein Beschreibungssatz nach DC erforderlich, der an dieser Stelle ebenfalls aus den o.g. Gründen nicht wiedergegeben wird.

Die erste Lerneinheit beinhaltet die Darstellung des Versuchsaufbaus "Vergasermotor-Generator-Satz", die zweite Lerneinheit ein Messprotokoll mit Anweisungen zur Aufnahme von Messwerten.

A-F-Matrix: Es wird das Attribut *Energieumsatz* mit der Funktion *Wandlung* gewählt. Damit wird dieser Lehrgang dem Anwendungsfeld *Versorgung und Entsorgung* zu-

geordnet. Das technische Sachsystem wird spezifiziert durch die Eintragung *Energiewandelungstechnik*.

Gegenstand: *Der Vergasermotor-Generator-Satz*

Thema: *Messen der energetischen Ein- und Ausgangsgrößen* (E_{chem}, E_{elektr}) *des Vergasermotor-Generator-Satzes zur erstmaligen Bestimmung des Nutzungsgrades* [Wehling, 1996].

Schwerpunktlernziel: *Der Nutzungsgrad des Vergasermotor-Generator-Satzes kann mit Hilfe der Messgrößen Benzinverbrauch, elektrische Spannung, elektrische Stromstärke und Zeit bestimmt werden* [Wehling, 1996].

Lernzieltaxonomie: Kognitive Dimension: Kenntnis; Affektive Dimension: Werten; Psychomotorische Dimension: Präzision

6.4 Sitemap

Neben den schon dargestellten Ausschnitten der Sitemap der *datenbankbasierten Entwicklungsumgebung* sei an dieser Stelle auf die dynamische Darstellung der methodischen Struktur durch ein Java-Applet verwiesen. Die nachfolgende Abbildung gibt einen ersten Eindruck, der jederzeit über die beiden URI http://www.mmdb-tu.de und http://www.integer-tu.de vertieft werden kann.

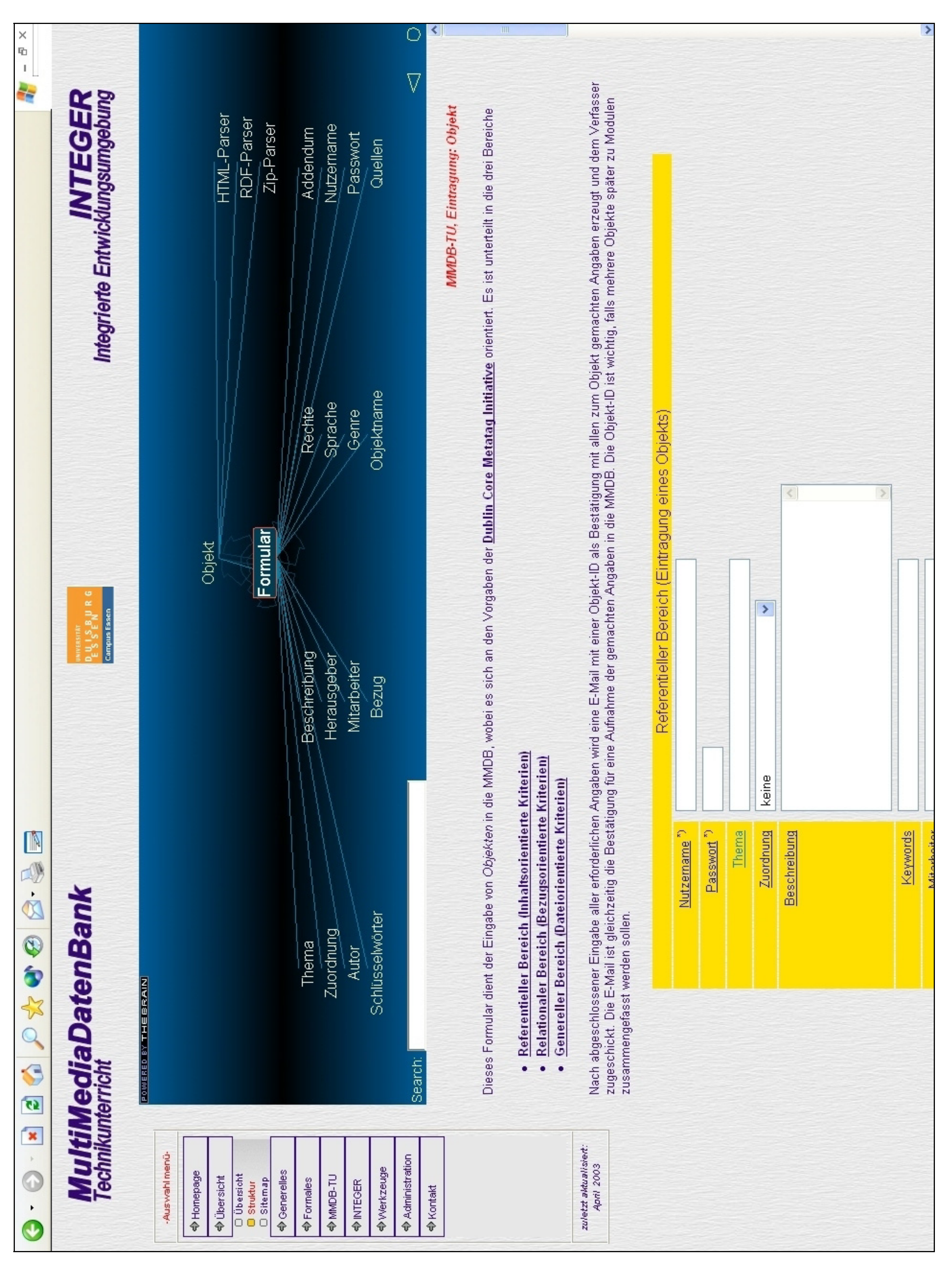


Abbildung 29: Struktur der *datenbankbasierten Entwicklungsumgebung*

6.5 Screenshots

Die folgenden Screenshots wurden bei einer Bildschirmauflösung von 1280x960 Pixel aufgenommen. Als Browser wurde der Internet Explorer 6.0 verwendet, als Betriebssystem Windows XP.

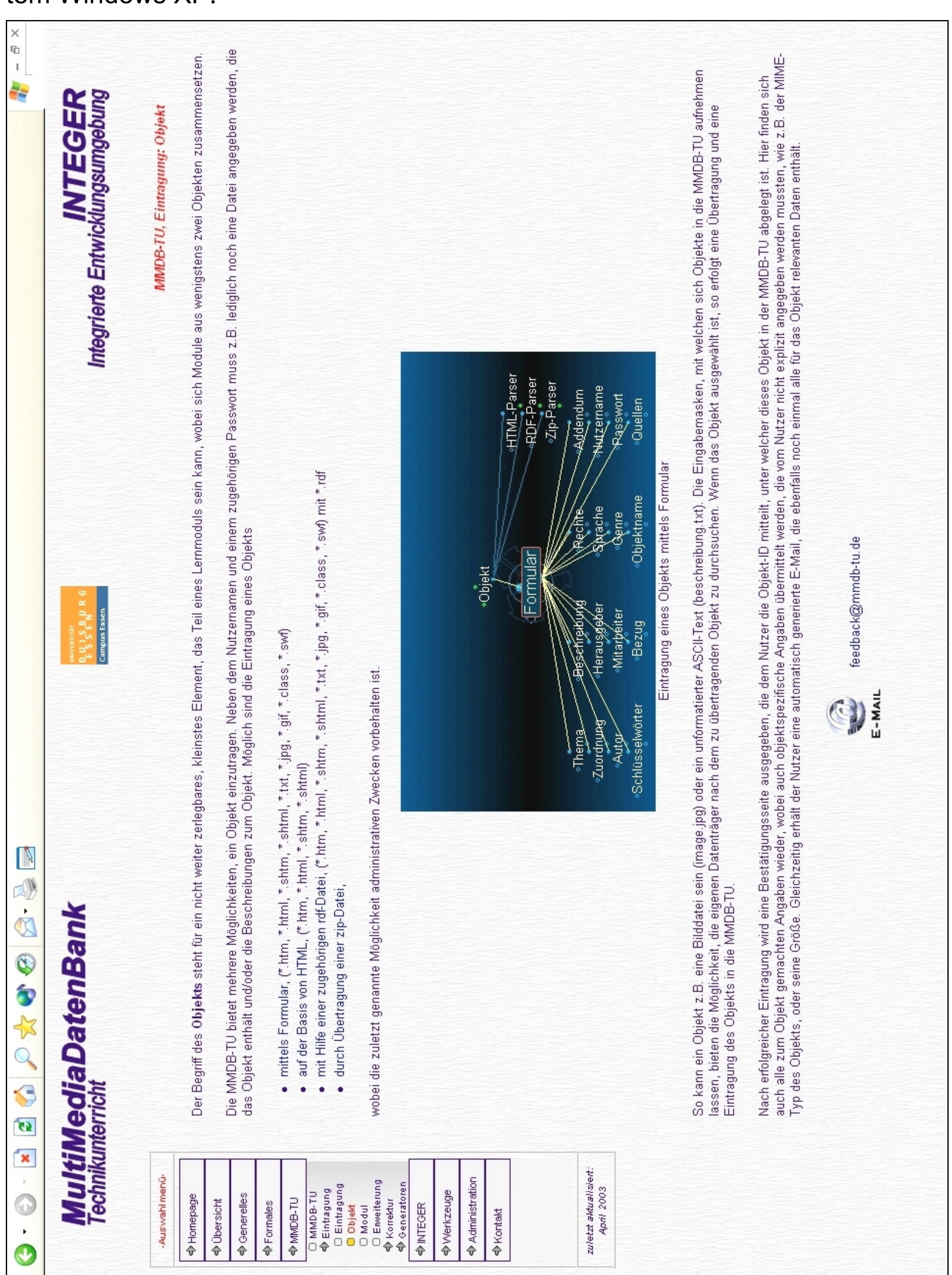


Abbildung 30: Screenshot: Eintragung eines Objekts

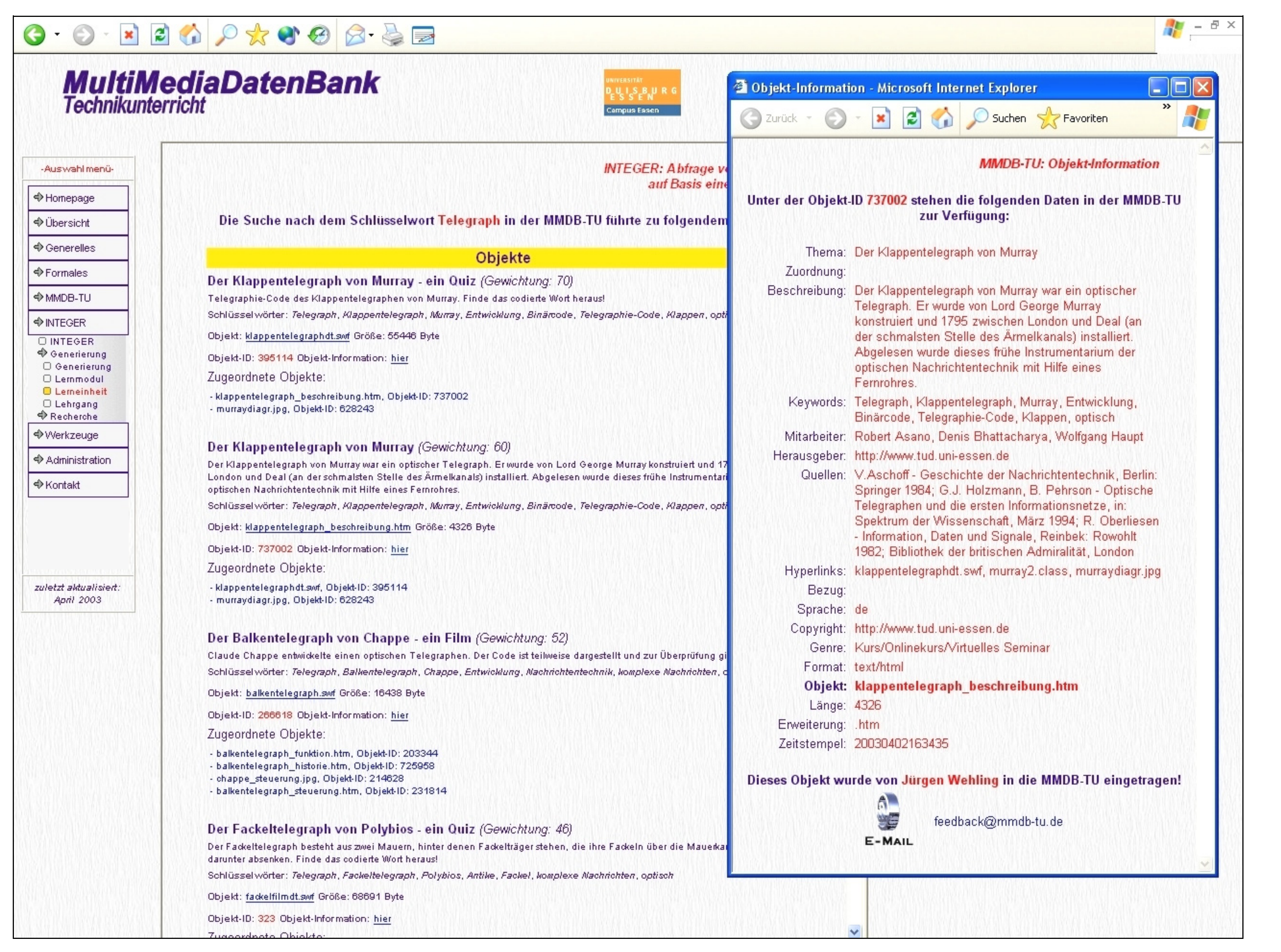


Abbildung 31: Screenshot: Objekt-Information

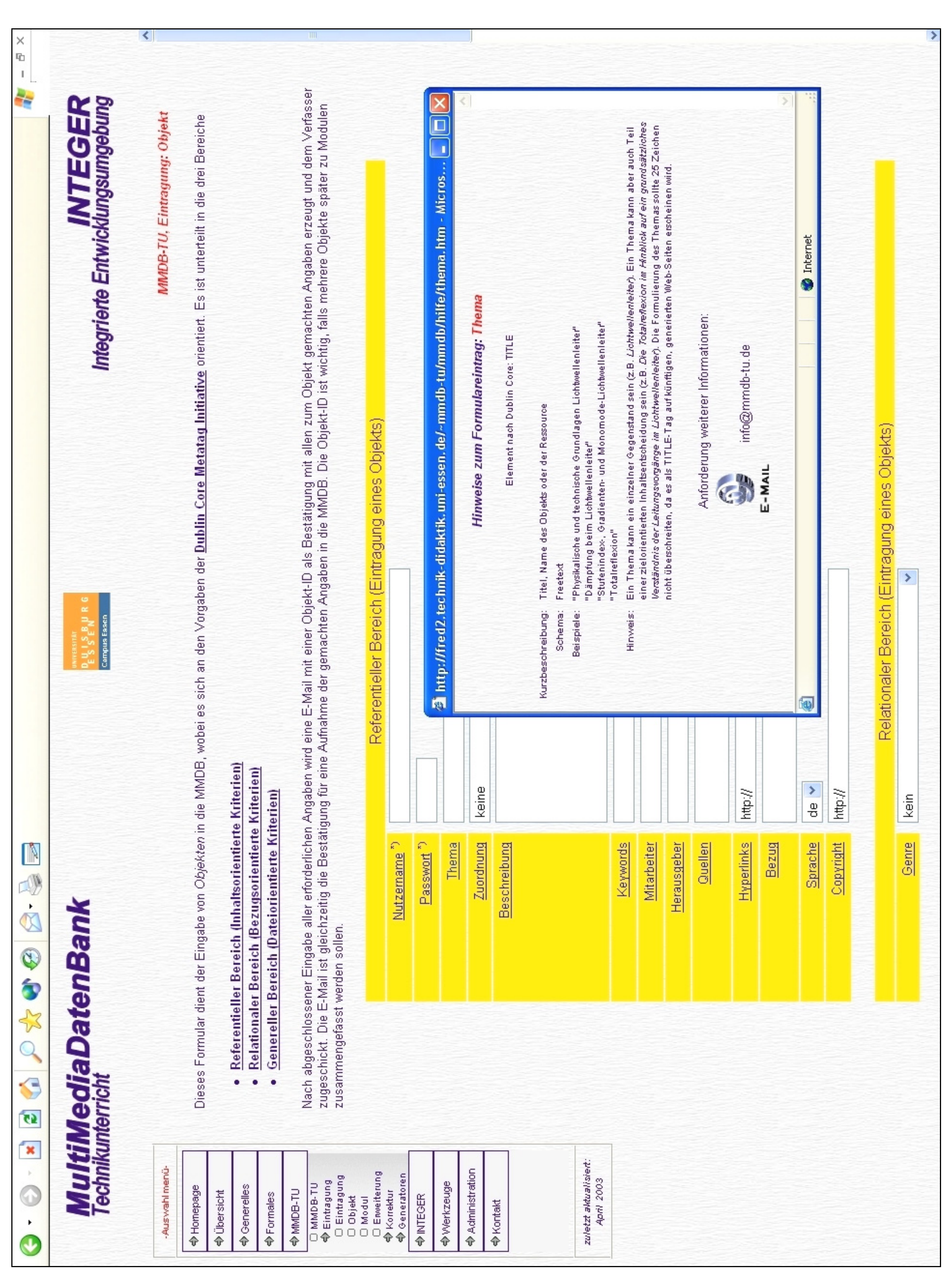


Abbildung 32: Screenshot: Integrierte Hilfefunktion

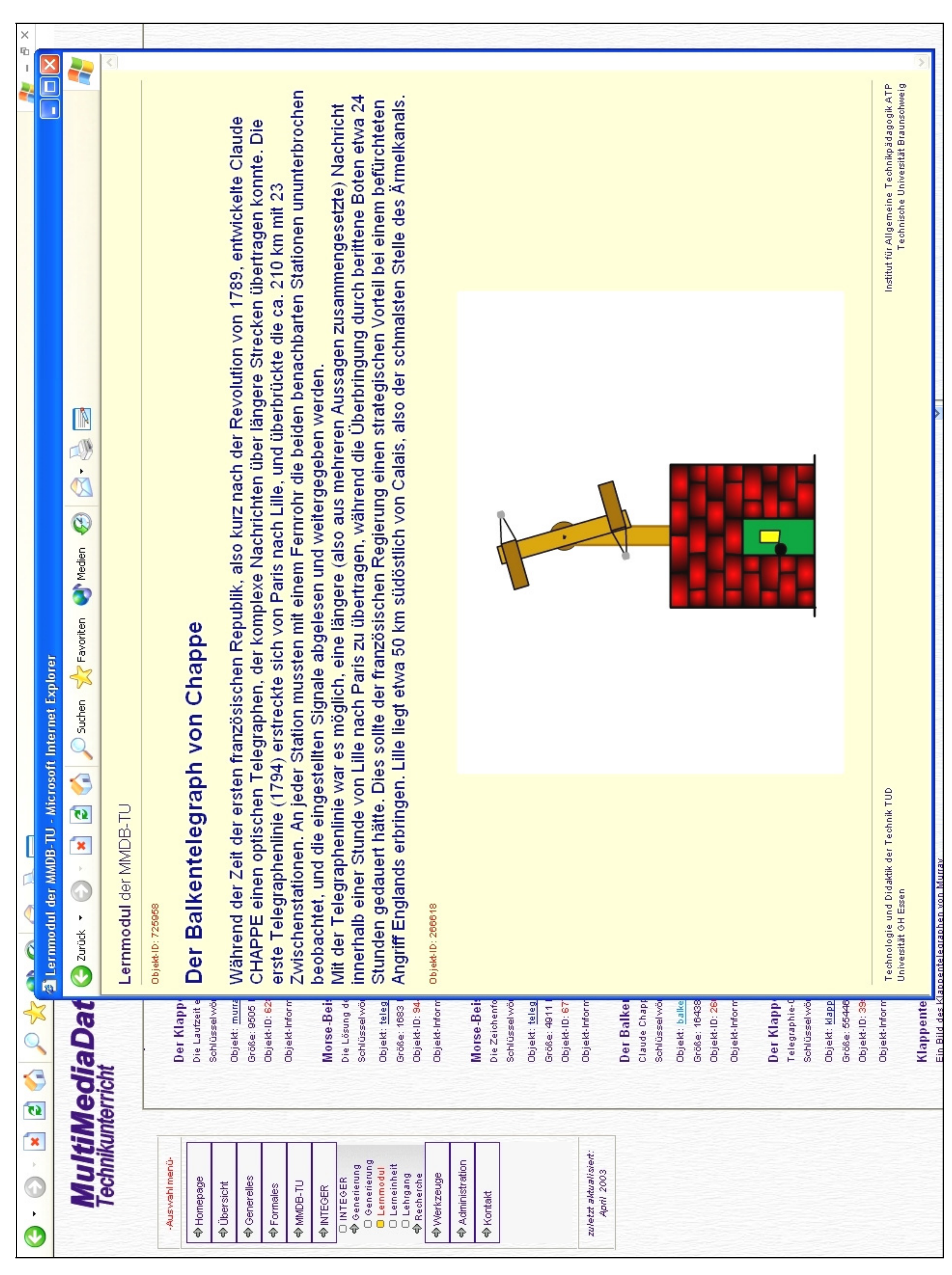


Abbildung 33: Screenshot: Neues Lernmodul

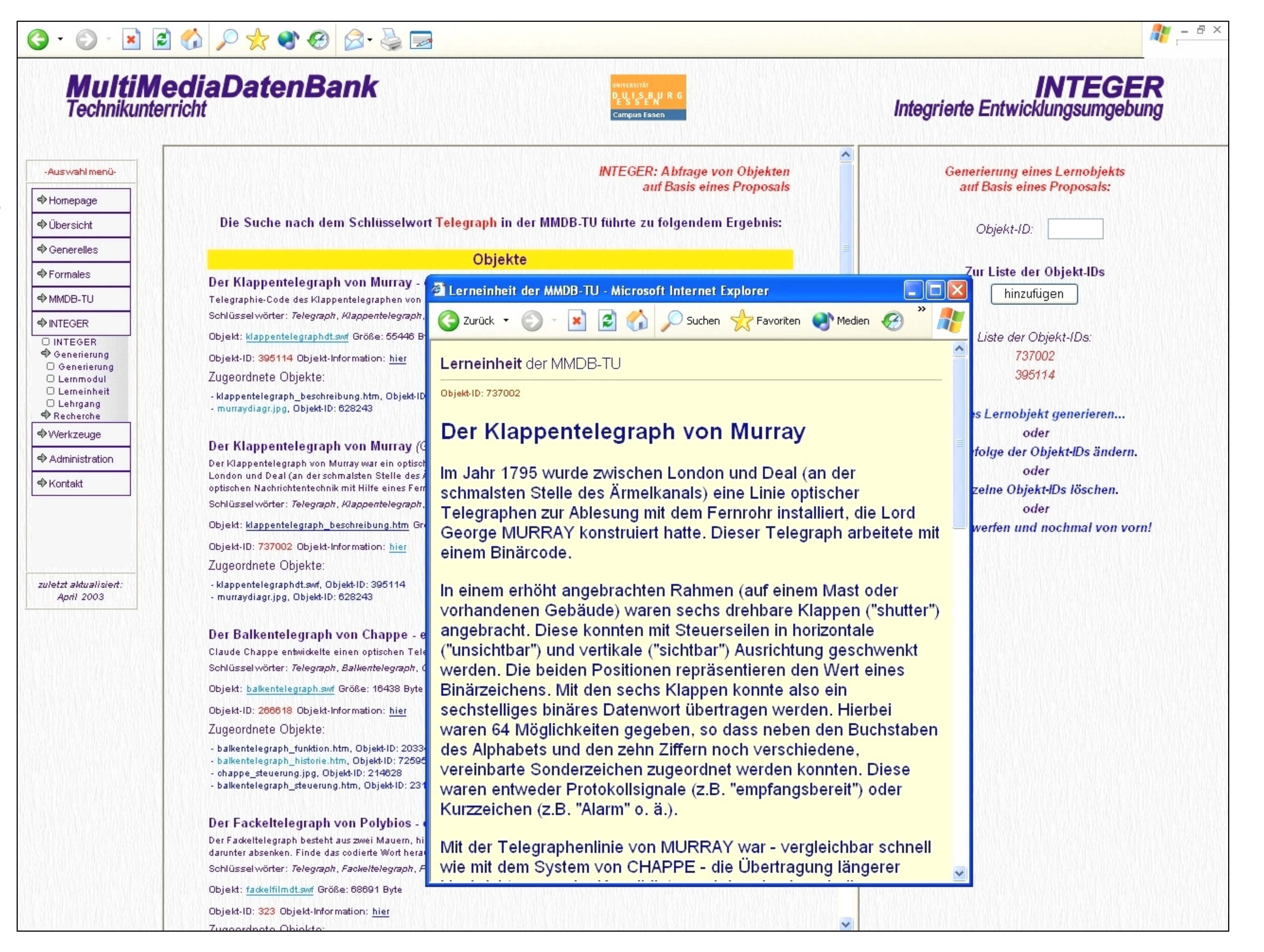


Abbildung 34: Screenshot: Neue Lerneinheit

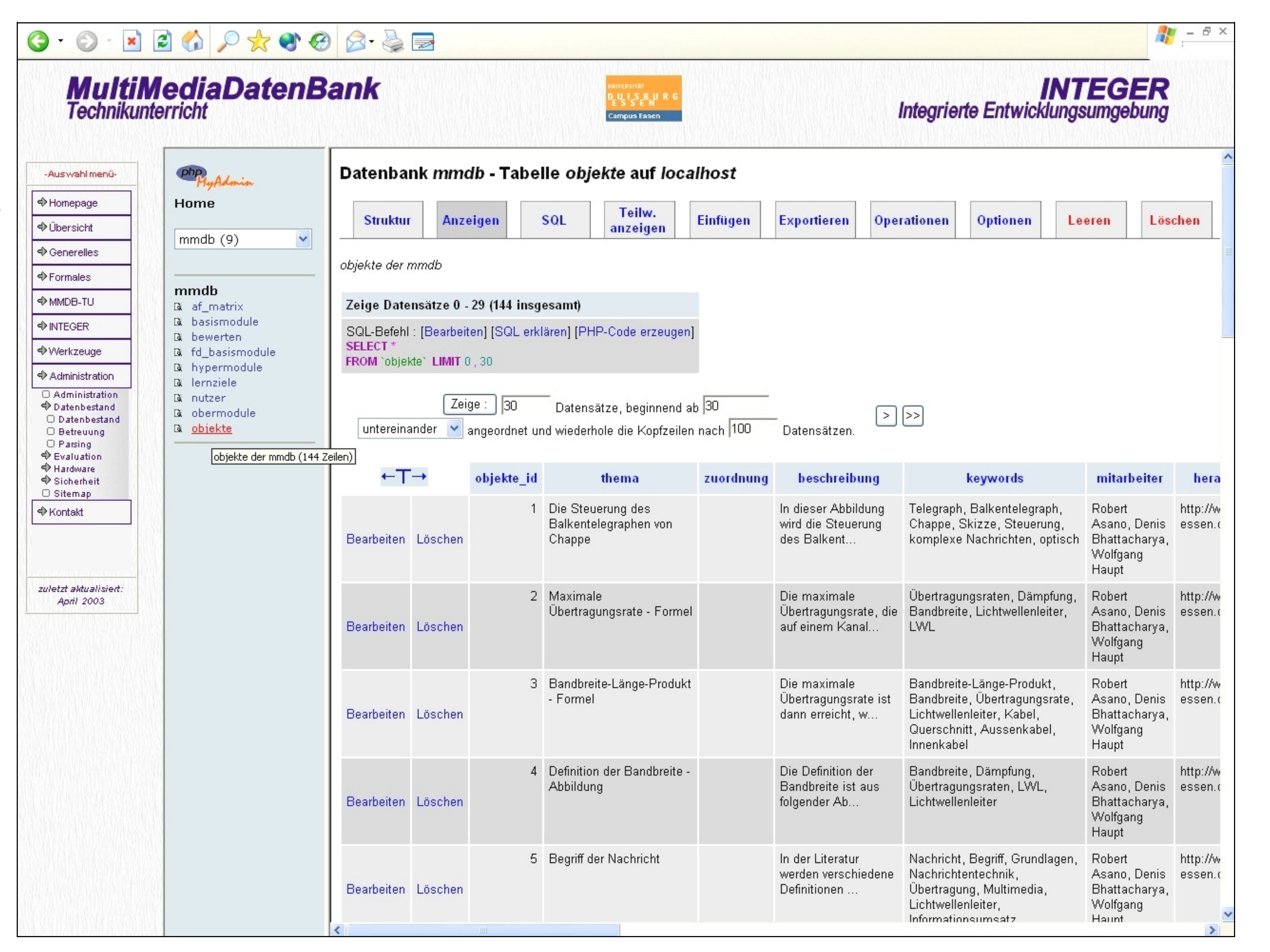


Abbildung 35: Screenshot: Administrativer Bereich, phpMyAdmin

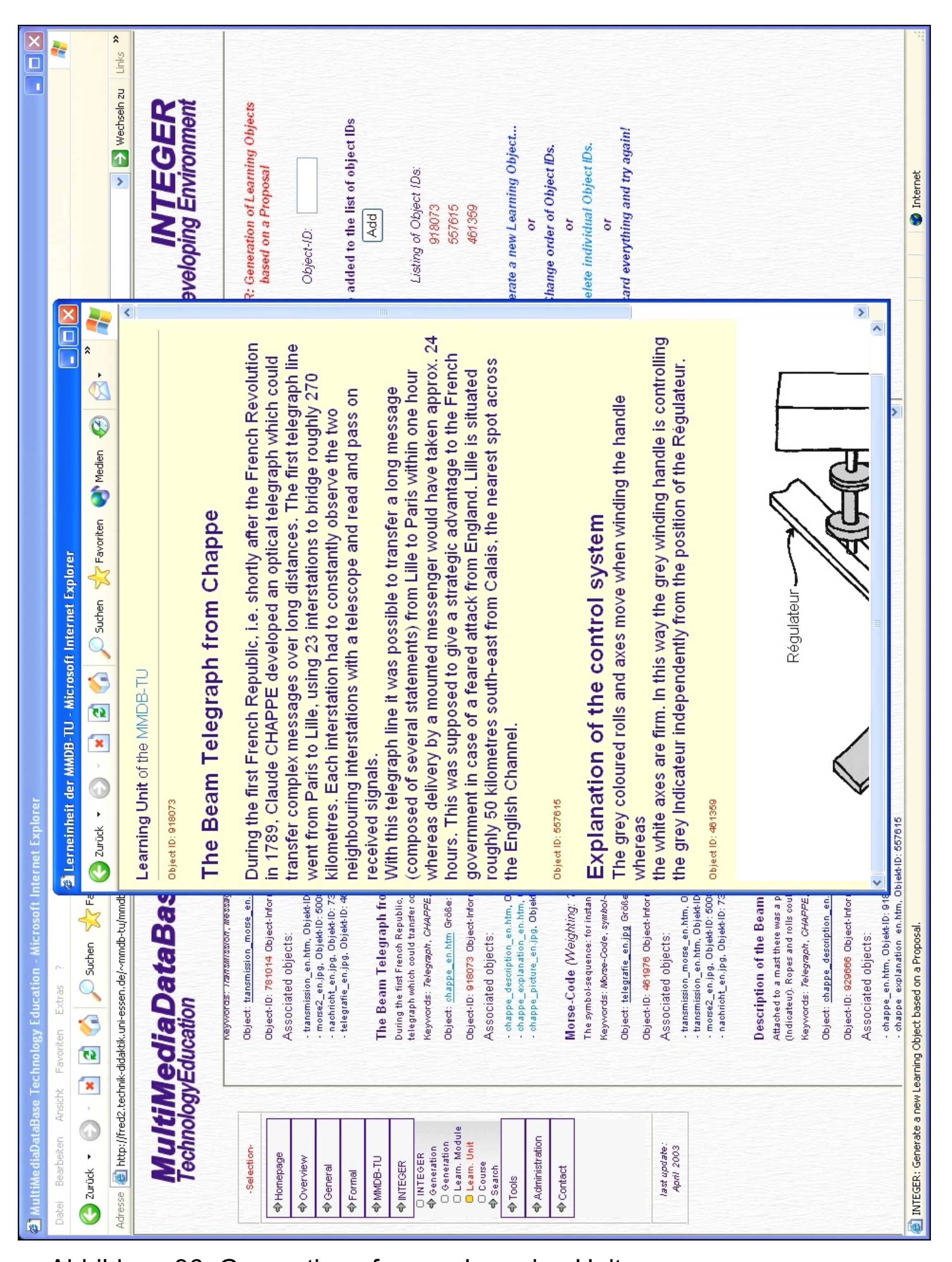


Abbildung 36: Generation of a new Learning Unit

Der internetbasierte Zugang zur englischen Version der datenbankbasierten Entwicklungsumgebung erfolgt ebenfalls durch die beiden URI http://www.mmdb-tu.de und http://www.integer-tu.de.

Literaturverzeichnis

Hinweis: Hyperlinks zu *Websites*, *Webpages* oder webgebundenen *pdf-Dokumenten* sind in der Regel kurzlebig. So kann ein noch kürzlich erfolgreich referenzierter Link schon in der nächsten Minute nicht mehr verfügbar sein. Um diesem Umstand vorzubeugen, wurden die damit verbundenen wesentlichen Inhalte zusätzlich dauerhaft verlinkt[1]. Ein solcher Link ist durch **lll** für **l**ong-**l**asting **l**ink gekennzeichnet.

[Abiturfach Technik, 2003] Technik als 3. oder 4. Abiturfach, *Webpage:* http://www.hg-essen.de/faecher/feld_III/technik-2/technik-2.htm, referenziert: 14.07.2003, lll

[ARIADNE, 2002] ARIADNE Foundation for the European Knowledge Pool. *Webpage:* http://www.ariadne-eu.org/en/system/index.html, referenziert: 14.07.2003, lll

[ARIADNE, 2002a] ARIADNE Educational Metadata Recommandation V3.2, *Webpage:* http://www.ariadne-eu.org/en/publications/metadata/ams_v32.html, referenziert: 14.07.2003, lll

[Asano, 2001] Asano, R./ Bhattacharya, D./ Wehling, J./ Haupt, W.: MultiMediaModul Lichtwellenleiter, *Website:* http://it.tud.uni-essen.de, referenziert: 14.07.2003, lll

[Aufenanger, 2002] Aufenanger, S.: Lernen mit neuen Medien – Mehr Wissen und bessere Bildung, Vortrag: education quality forum, Vortragsreihe: Wirkungen und Wirksamkeit Neuer Medien in der Bildung, 21./22.11.2002, *pdf-Dokument:* http://www.erzwiss.uni-hamburg.de/Personal/Aufenanger/Vortraege/Lernen mit neuen Medien Dortmund_Handzettel.pdf, referenziert: 14.07.2003, lll

[AWStats, 2003] AWStats, Advanced Webstatistics, *Webpage:* http://awstats.sourceforge.net, referenziert: 14.07.2003, lll

[Bader, 1975] Bader, R./ Haupt, W./ Heescher, H./ Hochstrate, G.-A./ Sanfleber, H./ Schulze-Fröhlich, D. F./ Thiele, W./ Wagener, W.: Curriculum Technik für die neu gestaltete gymnasiale Oberstufe in der Sekundarstufe II, Grundkurse, Fassung vom 10.10.1975, Archiv des Faches Technologie und Didaktik der Technik TUD

[Banke, 2002] Banke, K.: Erfahrungswerte, Open-Source-RDBMS von SAP, in: iX, Magazin für professionelle Informationstechnik, Ausgabe 08/2002, Hannover, Heise Verlag, 2003, ISSN: 0935-9680

[Baudry, 2002] Baudry A./ Bungenstock, M./ Mertsching, B.: Ein multimediales Rahmenwerk für die Mathematiklehre nach der Baukastenmetapher, in: Klaus P. Jantke, Wolfgang S. Wittig, Jörg Herrmann (Hrsg.): Von e-Learning bis e-Payment, Berlin 2002, Akademische Verlagsgesellschaft, AKA, ISBN 3-89838-033-5, S. 300-307.

[Baumgartner, 2002] Baumgartner, P./ Häfele, H./ Maier-Häfele M.: E-Learning Praxishandbuch, Auswahl von Lernplattformen, Innsbruck, Österreich, Studien Verlag, 2002, ISBN: 3-7065-1771-X

---

[1] Es erfolgt ein durch Nutzername und Passwort geschützer Zugang zu einem nicht für die Öffentlichkeit freigegebenem Verzeichnis des ftp-Servers des Faches TUD: ftp://fred3.technik-didaktik.uni-essen.de

[Baumgartner, 2002a] Baumgartner, P./ Häfele, H./ Maier-Häfele M.: E-Learning: Didaktische und technische Grundlagen, Sonderdruck, Perg: CD Austria Verlags- und Handelsgesellschaft mbH, 2002; als *pdf-Dokument:* http://iol3.uibk.ac.at:8080/filer/peterManilaWebsite/peter/material/e-learning_sonderheft.pdf, referenziert: 14.07.2003, III

[Bentlage, 2002] Bentlage, U.: Neue Allianzen: Kooperation zum wechselseitigen Nutzen, Bosten Consulting Group, Düsseldorf, 2002, *Website:* http://www.bcg.de, referenziert: 14.07.2003, III; Vortrag: education quality forum, Vortragsreihe: Wirkungen und Wirksamkeit Neuer Medien in der Bildung, 21./22.11.2002, Kongresszentrum Westfalenhallen Dortmund

[Bertelsmann, 2003] Bertelsmann Stiftung, Projektindex, *Webpage:* http://www.bertelsmann-stiftung.de/projects.cfm?lan=de&nId=33, referenziert: 14.07.2003, III

[Bhattacharya, 2001] Bhattacharya, D./ Wehling, J.: University teacher education for courses "Information Technology" as part of teacher education for Gymnasia secondary level with focus on the computer based modul "Optical Data Transmission", PATT-11 conference for New Media in Technology Education, Haarlem, Netherlands, Mottier/De Vries, 2001

[bildung.at, 2003] bildung.at, Content Management Systeme – Evaluationskriterien, *Webpage:* http://www.virtual-learning.at/cms/mindestkriterien.htm, referenziert: 14.07.2003, III

[Bildungsportal, 2003] BILDUNGSPORTAL.NRW, Ministerium für Schule, Wissenschaft und Forschung des Landes Nordrhein-Westfalen, *Webpage:* http://www.bildungsportal.nrw.de/BP/, referenziert: 19.07.2003, III

[Bilke, 2001] Bilke, P.: PHP und MySQL – dynamische Webseiten, 1. Auflage, Bonn, Presse Vertrieb, 2001, ISBN: 87-90785-70-3

[Blackboard, 2003] Blackboard Inc., the leading provider of Internet infrastructure software for e-Education, *Webpage:* http://ml.blackboard.com/worldwide/index.htm, referenziert: 14.07.2003, III

[BLBS, 2003] Bundesverband der Lehrerinnen und Lehrer an beruflichen Schulen e.V., Friedrichstr. 169/170, 10117 Berlin, Tel.: 030-40816650, Fax: 030-40816651, *Website:* http://www.blbs.de, referenziert: 14.07.2003, III

[Block, 1997] Block, K./ Hölzel, H.-J./ Wölfing, L./ Zachert, P.: Einführung in die Elektronik 2, Kontaktlose Signalverarbeitung, Köln, Stam Verlag, 1997, ISBN: 3-8237-0215-7

[Bloom, 1984] Bloom, B. S./ Krathwohl, D. R.: Taxonomy of Educational Objectives, Boston, Addison-Wesley Verlag, 1984, ISBN: 0-5822-8010-9

[Blue, 2003] Blue Titan Software, Web Services Control Layer, *Website:* http://www.bluetitan.com, referenziert: 14.07.2003, III

[Blumstengel, 1998] Blumstengel, A.: Entwicklung hypermedialer Lernsysteme, Berlin, Wissenschaftlicher Verlag Berlin, 1998, ISBN: 3-932089-13-8

[bm:bwk, 2003] Bundesministerium für Bildung, Wissenschaft und Kultur, Wien, Österreich, *Website:* http://www.bmbwk.gv.at/start.asp, referenziert: 14.07.2003, III

[BMBF, 2003] Förderung von Bildung und Forschung ist Zukunftsaufgabe, *Webpage:* http://www.bmbf.de/index_foerde01.htm, referenziert: 14.07.2003, III

[Bonsen, 2002] Bonsen, E./ Hey, G.: Kompetenzorientierung, *pdf-Dokument:* http://lehrplan.lernnetz.de/html/material/allgemein/kompetenzorientierung.pdf, referenziert: 14.07.2003, III

[Brand, 2002] Brand, M.: Phönix aus der Asche, Firebird 1.0: Das 'wahre' Interbase, in: iX, Magazin für professionelle Informationstechnik, Ausgabe 08/2002, Hannover, Heise Verlag, 2003, ISSN: 0935-9680

[c't, 2003] MS-E-Books schutzlos, in: c't, Magazin für Computertechnik, Ausgabe 02/2003, Hannover, Heise Verlag, 2003, ISSN: 0724-8679

[CMMV, 1999] Clearingstelle Multimedia für Verwertungsgesellschaften von Urheber- und Leistungsschutzrechten GmbH, *Website:* http://www.cmmv.de, referenziert: 14.07.2003, III

[CPAN, 2003] Comprehensive Perl Archive Network, CPAN, *Website:* http://www.cpan.org/, referenziert: 14.07.2003, III

[DAAD, 2003] Deutscher akademischer Austauschdienst e.V., Bildungsprogramme der Europäischen Union, *Website:* http://eu.daad.de, referenziert: 14.07.2003, III

[Darmstadt, 2003] Technische Universität Darmstadt, Glesner, M.: Online-Vorlesung, Mikroelektronische Systeme, *Webpage:* http://www.microelectronic.e-technik.tu-darmstadt.de/lectures/online/vhdl-online/main.htm, referenziert: 14.07.2003, III

[Date, 1998] Date, C./ Darwen, H.: SQL - Der Standard, SQL/92 mit den Erweiterungen CLI und PSM, 3. Auflage, München, Addison-Wesley Verlag, ISBN: 3-8273-1345-7

[Day, 2002] Day, M.: Metadata, mapping between metadata formats, UKOLN: The UK Office for Library and Information Networking, University of Bath, Bath, BA2 7AY, United Kingdom, *Webpage:* http://www.ukoln.ac.uk/metadata/interoperability/, referenziert: 14.07.2003, III

[DCdot, 2000] DCdot, Dublin Core Metadata Editor, UKOLN: The UK Office for Library and Information Networking, University of Bath, Bath, BA2 7AY, United Kingdom, *Webpage:* http://www.ukoln.ac.uk/metadata/dcdot/, referenziert: 14.07.2003, III

[DCMI, 2002] Dublin Core Metadata Initiative, *Website:* http://dublincore.org, referenziert: 14.07.2003, III

[DCMI, 2002a] Dublin Core Metadata Initiative: e-Learning takes important step forward, *Webpage:* http://dublincore.org/news/pr-20001206.shtml, referenziert: 14.07.2003, III

[DCMI, 2002b] Dublin Core Metadata Initiative: Expressing Qualified Dublin Core in RDF / XML, *Webpage:* http://dublincore.org/documents/2002/04/14/dcq-rdf-xml/sec1, referenziert: 07.04.2003, III

[DCMI, 2003] Dublin Core Metadata Initiative: DCMI Meta Data Terms, *Webpage:* http://dublincore.org/documents/dcmi-terms/H4, referenziert: 07.04.2003, III

[DCMIType, 2003] Dublin Core Metadata Initiative: DCMI Type Vocabulary, *Webpage:* http://dublincore.org/documents/dcmi-type-vocabulary/, referenziert: 14.07.2003, III

[Descartes, 2001] Descartes, A./ Bunce, T.: Programmierung mit Perl DBI, 1. Auflage, Köln, O'Reilly Verlag, 2001, ISBN: 3-89721-143-2

[Didszuns, 2003] Didszuns, R.: Internet-Glossar, Webpage: http://www.tigerweb.de/internet/glossar/lex1.htm, referenziert: 10.08.2003, III

[DigiBib, 2003] Digitale Bibliothek, Internet-Portal für den weltweiten Zugang zu wissenschaftlichen Informationen, *Website:* http://www1.digibib.net/, referenziert: 14.07.2003, III

[Dietrich, 2000] Dietrich, S. (Projektleitung): Projekt: Selbstgesteuertes Lernen (SeGeL), Deutsches Institut für Erwachsenenbildung e.V., gefördert vom BMBF, 2000, *pdf-Dokument:* http://www.die-bonn.de/segel/ergebnisse/segel_info_1.pdf, referenziert: 14.07.2003, III

[DIM, 2003] dim.digitale medien GbR, Treitschkestr. 31-32, D-12163 Berlin, Tel.: +49 30 39789683, Fax: +49 30 39789684, *Website:* http://www.dim-digitale-medien.de, referenziert: 07.02.2003, III

[DGTB, 2003] Deutsche Gesellschaft für Technische Bildung, *Website:* http://www.dgtb.de, referenziert: 12.07.2003

[Dohnke, 2002] Dohnke, H.: Lernziele, Studienseminar Sek II Recklinghausen, *Webpage:* http://www.learn-line.nrw.de/angebote/lakonkret/, referenziert: 14.07.2003, III

[Dubois, 2000] Dubois, P.: MySQL, Entwicklung, Implementierung, Referenz, 3. Auflage, München, Markt+Technik Verlag, 2000, ISBN: 3-8272-5793-X

[EasyProf, 2003] Powerful Multimedia Course Authoring, *Website:* http://www.easyprof.com/home/home.jsp, referenziert: 14.07.2003, III

[EGTB, 1999] Europäische Gesellschaft für Technische Bildung, *Webpage:* http://www.wocate.org/egtb/egtb.html, referenziert: 14.07.2003, III

[Encarnação, 2000] Encarnação, J. L./ Leidhold, W./ Reuter, A.: Die Universität im Jahre 2005, in: Informatik Spektrum, Organ der Gesellschaft für Informatik e.V., Ausgabe 08/2000, Berlin, Springer Verlag, 2000, ISSN: 0170-6012

[Energiewelten, 2000] Arbeitskreis Schulinformation Energie, Doppel-CD-ROM Energiewelten, Am Hauptbahnhof 12, 60329 Frankfurt, Tel.: +49 69 25619-148, Fax: +49 69 232721

[Enosh, 1999] Enosh, B.: Java and Databases, *pdf-Dokument:* http://www.cs.colorado.edu/~getrich/Classes/csci5817/Term_Papers/enosh/5817.pdf, referenziert: 14.07.2003, III

[Enseleit, 2001] Enseleit, D.: PHP 3/4, Befehlsreferenz, Poing, Franzis' Verlag, 2001, ISBN: 3-7723-7184-1

[EQ, 2002] education quality forum, Vortragsreihe: Wirkungen und Wirksamkeit Neuer Medien in der Bildung, 21./22.11.2002, Kongresszentrum Westfalenhallen Dortmund, *Website:* http://www.education-quality.de, referenziert: 02.02.2003, III

[Europa, 2003] Lebenslanges Lernen, Europäische Kommission, *Webpage:* http://europa.eu.int/comm/education/lll_de.html, referenziert: 02.02.2003, III

[Fachkunde Metall, 1999] Fachkunde Metall, Haan-Gruiten, Europa Verlag, 1999, ISBN: 3-8085-1153-2

[Falz, 2002] Falz, S./ Samaschke, K.: Das ASP Codebook, München, Addison-Wesley Verlag, 2002, ISBN: 3-8273-1948-X

[Ferber, 2003] Ferber, R.: Information Retrieval, Suchmodelle und Data-Mining-Verfahren für Textsammlungen und das Web, Heidelberg, dpunkt Verlag, 2003, ISBN: 3-89864-213-5, S. 276

[Ferber, 2003a] Ferber, R.: Information Retrieval, Suchmodelle und Data-Mining-Verfahren für Textsammlungen und das Web, Heidelberg, dpunkt Verlag, 2003, ISBN: 3-89864-213-5, S. 268-272

[FILExt, 2003] FILExt, The file extension source, *Website:* http://www.filext.com/, referenziert: 14.07.2003, III

[Finsterle, 2002] Finsterle, L./ Rotard, M.: Mit konventionellen Autorensystemen zum E-Learning Portal, in: Klaus P. Jantke, Wolfgang S. Wittig, Jörg Herrmann (Hrsg.): Von e-Learning bis e-Payment, Berlin 2002, Akademische Verlagsgesellschaft, AKA, ISBN 3-89838-033-5, S. 111-121

[Fittkau, 2003] Fittkau & Maaß, W3B Browserwatch, *Webpage:* http://www.w3b.org/trends/browserwatch.html, referenziert: 14.07.2003, III

[Flanagan, 2002] Flanagan, D.: JavaScript, Das umfassende Referenzwerk, Köln, O-Reilly, 2002, ISBN: 3-8972-1330-3

[Fraunhofer, 2001] Fraunhofer Institut, Integrierte Publikations- und Informationssysteme, MPEG7, *Webpage:* http://ipsi.fraunhofer.de/delite/Projects/MPEG7/, referenziert: 14.07.2003, III

[Fraunhofer, 2002] Fraunhofer Gesellschaft: Multimediarecht in Wissenschaft und Forschung, 2002, *pdf-Dokument:* http://www.zi.fh-koeln.de/neu/service/multimedia/UrhG/Hoeren_UrhG_9_02.pdf, referenziert: 14.07.2003, III

[Friedl, 2001] Friedl, J. E. F.: Reguläre Ausdrücke, 2. korrigierter Nachdruck, Köln, O-Reilly Verlag, 2001, ISBN: 3-930673-62-2

[Geller, 2003] Geller, W.: Thermodynamik für Maschinenbauer, Grundlagen für die Praxis, Berlin, Springer Verlag, 2003, ISBN: 3-540000-305

[Geschwinde, 2002] Geschwinde, E./ Schönig, H.-J.: Datenbank-Anwendungen mit PostgreSQL, Einführung in die Programmierung mit SQL, Java, C/C++, Perl, PHP, München, Markt+Technik Verlag, 2002, ISBN: 3-8272-6394-8

[GMW, 2003] 8. Europäischer Kongress der Gesellschaft für Medien in der Wissenschaft, 16.-19.09.2003 Universität Duisburg, *Website:* http://www.gmw03.de, referenziert: 14.07.2003, III

[Goller, 2002] Goller, H.: Das LAMP Buch. Webserver mit Linux, Apache, MySQL und PHP, 1. Auflage, Nürnberg, SuSE Press, 2002, ISBN: 3-935922-09-4

[Graßmuck, 1994] Graßmuck, J./ Houben, K.-W./ Zollinger, R.: DIN-Normen in der Verfahrenstechnik, 2. neubearbeitete Auflage, Stuttgart, Teubner Verlag, 1994, ISBN: 3-519-16332-2, S. 36-46

[Graube, 2002] Graube, G./ Theuerkauf, W.:Datenbanken zur Technischen Bildung – Konzept zur Verbesserung der Qualität der Lehre, in:Technische Bildung, Ansätze und Perspektiven, 1. Auflage, Frankfurt am Main, Lang Verlag, 2002, ISBN 3-631-38548-X, S. 267-274

[Groten, 2002] Groten, H.: Universitätsverbund Multimedia, Hagen, in: education quality forum, Wirkungen und Wirksamkeit Neuer Medien in der Bildung, 21./22.11.2002, Kongresszentrum Westfalenhallen Dortmund, *Website:* http://www.education-quality.de, referenziert: 19.07.2003, III

[Gudjons, 2002] Gudjons, H. (Hrsg.): Didaktische Theorien, 2002, Hamburg, Bergmann + Helbig Verlag, ISBN: 3-9258-3635-7

[Guilford, 1971] Guilford, J. P.: A system of psychomotor abilities, in: American Journal of Psychology, University of Illinois Press, Champaign, USA, 1971, IL 61820-6903, S. 164-174

[Haack, 1997] Haack, J.: Interaktivität als Kennzeichen von Multimedia und Hypermedia. in: Issing, L./ Klimsa, P. (Hrsg.): Information und Lernen mit Multimedia, 2. überarbeitete Auflage; S. 151-165; Weinheim, Basel, Beltz Psychologie-Verlags-Union, 1997, ISBN: 3-6212-7449-9

[Hagen, 2003] FernUniversität Gesamthochschule in Hagen, *Website:* http://www.fernuni-hagen.de, referenziert: 14.07.2003, III

[Hartwig, 2001] Hartwig, J.: PostgreSQL – professionell und praxisnah, München, Addison-Wesley Verlag, 2001, ISBN: 3-8273-1860-2

[Haupt, 1976] Haupt, W./ Sanfleber, H.: Ansatz einer Didaktik der Technik für den Technikunterricht der Sekundarstufe II (differenzierte gymnasiale Oberstufe), in: Traebert, H. E./ Spiegel, H. R. (Hrsg.), Technik als Schulfach, Düsseldorf, 1976, S. 185-204

[HEA, 2003] Fachverband für Energie-Marketing und –Anwendung e.V. beim VdEW, *Website:* http://www.hea.de, referenziert: 14.07.2003, III

[Heimann, 1979] Heimann, P./ Otto, G./ Schulz, W.: Unterricht. Analyse und Planung, Hannover, Schroedel Verlag, 1979, ISBN: 3-5073-6310-0

[Heise, 2001] Heise Online: Dot.coms beenden Jahr der Pleiten, *Webpage:* http://www.heise.de/newsticker/data/em-27.12.01-002/, referenziert: 14.07.2003, III

[Henning, 2001] Henning, P. A.: Taschenbuch Multimedia, 2. Auflage, München, Hanser Verlag, 2001, ISBN: 3-446-21751-7

[Hesselmann, 1987] Hesselmann, N.: Digitale Signalverarbeitung, 2. Auflage, Würzburg, Vogel Verlag, 1987, ISBN: 3-8023-0707-0

[Hillmann, 2001] Hillmann, D.: Using Dublin Core, *Webpage:* http://dublincore.org/documents/usageguide/, referenziert: 14.07.2003, III

[HIS, 2003] Hochschul-Informations-System GmbH, Arbeitsgebiet 17: Neue Medien-Nutzungs-, Planungs- und Organisationskonzepte, *Webpage:* http://www.his.de/His/JAP/planung_5_2, referenziert: 14.07.2003, III

[Holzinger, 2001] Holzinger, A.: Interoperabilität und Metadaten, Workshop am 2. Business Meeting “Forum Neue Medien“, Wien, 2001, *pdf-Dokument:* http://serverprojekt.fh-joanneum.at/sp/thema/meta/metadaten.pdf, referenziert: 14.07.2003, III

[Hübsch, 2003] Hübsch, C.: Wissensvermittlung im Baukastensystem, E-Learning und Classic-Learning mit modularen Lehrmaterialien, in: TU-Spektrum, Magazin der TU-Chemnitz, Ausgabe 4/2003, Chemnitz, PrintDesign GmbH Chemnitz, 2003, ISSN: 0946-1817, S. 27, alternativ: http://www.tu-chemnitz.de/spektrum/02-4/seiten/seite27.htm, referenziert: 01.06.2003, III

[IANA, 2003] Internet Assigned Numbers Authority, Port Numbers, *Webpage:* http://www.iana.org/assignments/port-numbers, referenziert: 14.07.2003, III

[IBM, 2003] International British Machines, *Website:* http://www.ibm.com, referenziert: 02.02.2003, III

[IEEE, 2003] Institute of Electrical and Electronic Engineers, Learning Technology Standards Committee, IEEE 1484.12 LOM-Working Group, *Webpage:* http://ltsc.ieee.org/wg12/index.html, referenziert: 26.03.2003, III

[IETF, 1995] RFC 1766: Tags for the Identification of Languages, in: The Internet Engineering Taskforce, *Webpage:* http://www.ietf.org/rfc/rfc1766.txt, referenziert: 07.04.2003, III

[IETF, 1999] RFC 2413, RFC 2731: Encoding Dublin Core Metadata in HTML, in: The Internet Engineering Taskforce, *Webpages:* http://www.ietf.org/rfc/rfc2413.txt?number=2413, referenziert: 21.02.2003, III, und http://www.ietf.org/rfc/rfc2731.txt?number=2731, referenziert: 21.02.2003, III

[ILIAS, 2003] Integriertes Lern-, Informations- und Arbeitskooperationssystem ILIAS, Köln, *Webpage:* http://www.ilias.uni-koeln.de/ios/, referenziert: 14.07.2003, III

[ILIAS, 2003a] Informationsseite des Integrierten Lern-, Informations- und Arbeitskooperationssystems ILIAS, Köln, *Webpage:* http://www.ilias.uni-koeln.de/ios/info.html, referenziert: 14.07.2003, III

[INTEGER, 2003] Integrierte Entwicklungsumgebung für die Generierung von Lernobjekten, Jürgen Wehling, *Website:* http://www.integer-tu.de, referenziert: 14.07.2003, III

[Intel, 2003] Intel Corporation, *Website:* http://www.intel.com, referenziert: 14.07.2003, III

[Interoperabilität, 2001] Interoperabilität und Metadaten, Themenworkshop am Wirtschaftsförderungsinstitut, Wien, 2001, *pdf-Dokument:* http://serverprojekt.fh-joanneum.at/sp/thema/meta/metadat-wsp.pdf, referenziert: 14.07.2003, III

[Iridium, 2003] Satelliten-Telefonnetz Iridium, *Website:* http://www.pulsar.ch/pulsar/sat-phones/home.htm, referenziert: 14.07.2003, III

[ITACA, 2003] ITACA, Rambla Brasil 38-40 local 3, Barcelona Catalunya, Spain, 08028, Phone: +34934906410, Fax: +34934906468, Contact: Eneko Arriaga, pm@easyprof.com

[IUM, 2003] Institut für Urheber- und Medienrecht e.V., Verabschiedung der Urheberrechtsreform verzögert sich weiter, München, *Webpage:* http://www.urheberrecht.org/news/?id=1295&w=&p=1, referenziert: 14.07.2003, III

[Jakupec, 2000] Jakupec, V./ Garrick, J.: Flexible Learning, Human Resource and Organisational Development, London, Routledge Verlag, 2000, ISBN: 0-4152-0060-1

[Joanneum, 2002] FH Joanneum, Serverprojekt 2002, Newsletter August 2002 – Neue Medien in der Lehre, *pdf-Dokument:* http://serverprojekt.fh-joanneum.at/noflash/new/archiv/august2002.pdf, referenziert: 14.07.2003, III

[Joerger, 1989] Joerger, K.: Einführung in die Lernpsychologie, 13. Auflage, Freiburg, Herder Verlag, 1989, ISBN: 3-4510-9043-0

[Junker, 1999] Junker, M.: Urheberrechtliche Probleme beim Einsatz von Multimedia und Internet in Hochschulen, Teil 1, in: JurPC Web-Dok. 69/1999, Internet-Zeitschrift für Rechtsinformatik, 1999, *Webpage:* http://www.jura.uni-sb.de/jurpc/aufsatz/19990069.htm, referenziert: 14.07.2003, III;Teil 2, in: JurPC Web-Dok. 86/1999, Internet-Zeitschrift für Rechtsinformatik, 1999, *Webpage:* http://www.jura.uni-sb.de/jurpc/aufsatz/19990086.htm, referenziert: 14.07.2003, III

[Kerres, 2001] Kerres, M.: Multimediale und telemediale Lernumgebungen, 2. Auflage, München, Oldenbourg Verlag, 2001, ISBN: 3-486-25055-8

[Kirsch, 2003] Kirsch, C.: Aufpoliert, Neuerungen in MySQL 4, in: iX, Magazin für professionelle Informationstechnik, Ausgabe 02/2003, Hannover, Heise Verlag, 2003, ISSN: 0935-9680

[Klafki, 1995] Klafki, W.: Studien zur Bildungstheorie und Didaktik (Book on Demand), Basel, Beltz Verlag, 1995, ISBN: 3-4071-8092-6

[Köster, 2002] Köster, M. L.: MK-Consulting, Code Red, *Webpage:* http://www.mkcsoftware.de/fhome.html?codered.html&2, referenziert: 12.07.2003

[Kofler, 2001] Kofler, M.: MySQL, Einführung, Programmierung, Referenz, München, Addison-Wesley Verlag, 2001, ISBN: 3-8273-1762-2

[Kraemer, 2001] Kraemer, W./ Müller, M. (Hrsg.): Corporate Universities und E-Learning, 1. Auflage, Wiesbaden, Gabler Verlag, 2001, ISBN: 3-4091-1855-1

[Krause, 2000] Krause, J.: PHP, Webserver-Programmierung unter Windows und Linux, München, Hanser Verlag, 2000, ISBN: 3-446-21301-5

[Krathwohl, 1978] Krathwohl, D. R./ Bloom, B. S./ Masia, B. B.: Taxonomie von Lernzielen im affektiven Bereich, 2. Auflage, Weinheim, Beltz u. Gelberg Verlag, 1978, ISBN: 3-4075-1085-3

[Kroner, 1997] Kroner, B./ Schauer, H.: Unterricht erfolgreich planen und durchführen, 1. Auflage, Köln, Aulis Verlag, 1997, ISBN: 3-7614-1924-4

[Landtag NRW, 2003] Landtag Nordrhein Westfalen, Drucksache 13/700, *Webpage:* http://home.landtag.nrw.de/mdl/ralf.witzel/d30.htm, referenziert: 14.07.2003, III

[Laurence, 1999] Laurence, S./ Giles, C. L.: Accessibility of information on the web, Nature, Volume 400, Nature Publishing Group, Division of Macmillan Publishers Limited, of Brunel Road, Houndmills, Basingstoke, Hampshire, England, 1999, RG21 6XS, Registered Number 785998

[Lit, 2002] Langkau, T./ Rudolph, C./ Wehling, J./ Haupt, W.: Multimediales Lernen im Fach Technologie und Didaktik der Technik, in: Klaus P. Jantke, Wolfgang S. Wittig, Jörg Herrmann (Hrsg.): Von e-Learning bis e-Payment, Berlin 2002, Akademische Verlagsgesellschaft, AKA, ISBN 3-89838-033-5, S. 276-289

[Lorenz, 2002] Lorenz, P. A.: ASP.Net, Grundlagen und Profiwissen, München, Hanser-Verlag, 2002, ISBN: 3-4462-1943-9

[LPF, 2003] Lernplattform ILIAS, *Webpage:* http://lpf.tud.uni-essen.de/ilias/, referenziert: 14.07.2003, III

[LPO, 2003] Ordnung der Ersten Staatsprüfungen für Lehrämter an Schulen, Lehramtsprüfungsordnung LPO, NRW, Stand: 27.03.2003, *pdf-Dokument:* http://www.zfl.uni-bielefeld.de/gesetze/lpo/lpo_2003.pdf, referenziert: 14.07.2003, III

[Lund, 1998] Dublin Core Metadata Template, Nordic Metadata Project, Lund University, Helsingborg, Schweden, *Webpage:* http://www.lub.lu.se/cgi-bin/nmdc.pl, referenziert: 14.07.2003, III

[Mager, 1994] Mager, R. F.: Lernziele und Unterricht, Weinheim, Beltz Verlag, 1994, ISBN: 3-4072-5156-4

[Matthiessen, 2000] Matthiessen, G./ Unterstein, M.: Relationale Datenbanken und SQL, München, Addison-Wesley Verlag, 2000, ISBN: 3-8273-1558-1

[Maslakowski, 2000] Maslakowski, M.: MySQL in 21 Tagen, München, Markt+Technik Verlag, 2000, ISBN: 3-8272-5850-2

[MetatagsDC, 2003] Metatag-Generator nach Dublin Core, MultiMediaDatenBank TechnikUnterricht, *Webpage:* http://fred2.technik-didaktik.uni-essen.de/~mmdb-tu/mmdb/dc_metatag.htm, referenziert: 14.07.2003, III

[Meyer, 2003] Meyer, A./ Schneller, A.: Oben auf, Informatikstudium und Artverwandtes, in: c't, Magazin für Computertechnik, Ausgabe 05/2003, Hannover, Heise Verlag, 2003, ISSN: 0724-8679

[Microsoft, 2003] Microsoft Corporation, *Website:* http://www.microsoft.com, referenziert: 14.07.2003, III

[Micsik, 1999] Micsik, A.: A study of portability in the deployment of WWW, in Acta Cybernetica, Volume 14, Issue 2, 1999, ISSN: 0324-721X

[MILESS, 2002] Multimedialer Lehr- und Lernserver Essen, Die Essener Digitale Bibliothek, *Website:* http://miless.uni-essen.de/, referenziert: 19.07.2003, III

[Mischke, 2002] Mischke, J: Ich bin so frei, MySQL und PostgreSQL im Vergleich, in: iX, Magazin für professionelle Informationstechnik, Ausgabe 01/2002, Hannover, Heise Verlag, 2003, ISSN: 0935-9680

[MMCL, 2003] Multimedia Clearingstelle der Universität GH Essen, *Website:* http://www.uni-essen.de/multimedia/, referenziert: 19.07.2003, III

[MMDB-TU, 2003] MultiMediaDatenBank TechnikUnterricht, Allgemeines, *Webpage:* http://fred2.technik-didaktik.uni-essen.de/~mmdb-tu/mmdb/allgemeines.htm, referenziert: 19.07.2003, III

[MSWF, 2003] Ministerium für Schule, Wissenschaft und Forschung, Bildungsportal NRW, Bestellung und Download von kostenlosen Broschüren, Update-Ratgeber Multimediarecht für die Hoschschulpraxis, *Webpage:* http://www.bildungsportal.nrw.de/BP/Service/broschueren/, referenziert: 19.07.2003, III

[Münz, 2002] Münz, S./ Nefzger, W.: HTML & Web-Publishing Handbuch, CGI 1.0/1.1 – Perl: Funktionen und Module – XML und DTDs – XML-Daten darstellen, Poing, Franzis' Verlag, 2002, ISBN: 3-7723-7515-4, S. 475

[Münz, 2002a] Münz, S./ Nefzger, W.: HTML & Web-Publishing Handbuch, HTML 4.01 – XHTML 1.0/1.1 – JavaScript – DHTML – CSS, Poing, Franzis' Verlag, 2002, ISBN: 3-7723-7515-4, S. 129

[Multimediarecht, 2001] Update-Ratgeber Multimediarecht für die Hochschulpraxis, Ministerium für Schule, Wissenschaft und Forschung des Landes Nordrhein-Westfalen, Völklinger Straße 49, 40221 Düsseldorf, Tel.: +49 211 89603, Fax: +49 211 8964514

[MySQL-Front, 2003] MySQL-Front Forum, Ventura Media, *Website:* http://mysqlfront.venturemedia.de/, referenziert: 19.07.2003, III

[MZ, 2003] Medienzentrum der Universität Essen, *Webpage:* http://www.mz.uni-essen.de/c1p3.html, referenziert: 19.07.2003, III

[Niegemann, 2001] Niegemann, H. M.: Neue Medien; Konzipieren, Entwickeln, Einsetzen; Bern, Schweiz, Huber Verlag, 2001, ISBN: 3-4568-3448-9

[Nixdorf, 2003] Hochschulnetzwerk Lehrerausbildung und Neue Medien, Heinz Nixdorf Stiftung, *Website:* http://www.lehrerbildung-medien.de/nixdorf.html, referenziert: 19.07.2003, III

[North, 1999] North, K.: Java in the Database, *Webpage:* http://archive.devx.com/upload/free/features/javapro/1999/03mar99/kn0399/kn0399.asp, referenziert: 19.07.2003, III

[OpenUSS, 2003] University Support System, *Website:* http://www.openuss.de, referenziert: 19.07.2003, III

[PATT-11, 2001] Wehling, J./ Bhattacharya, D.: University teacher education for courses "Information Technology" as part of teacher education for Gymnasia secondary level, PATT-11 (Pupils' Attitude Towards Technology) conference for New Media in Technology Education, Haarlem, Netherlands, Mottier/De Vries (Hrsg.), 2001

[PATT-12, 2002] Wehling, J./ Haupt, W.: Data structuring of online-based courses for technology education, PATT-12 (Pupils' Attitude Towards Technology) conference for Optimal Use of Resources in Technology Education, Kapstadt, Südafrika, Beute (Hrsg.), 2001

[PATT-12, 2002a] Haupt, W./ Wehling, J.: An online based learning module: "Optical Wave Guide" in technology education, PATT-12 (Pupils' Attitude Towards Technology) conference for Optimal Use of Resources in Technology Education, Kapstadt, Südafrika, Beute (Hrsg.), 2001

[phpMyAdmin, 2003] The phpMyAdmin Project, *Website:* http://www.phpmyadmin.net/, referenziert: 26.02.2003, III

[phpResource, 2003] The php Resource developer community, *Webpage:* http://www.php-resource.de/manual.php?p=history, referenziert: 02.03.2003, III

[Physics4You, 2003] Lernsoftware selbst gestalten, *Website:* http://www.physics4you.de/de/org/module.html, referenziert: 19.07.2003, III

[Potsdam, 2002] Universität Potsdam, Institut für Arbeitslehre/Technik, *Website:* http://www.uni-potsdam.de/u/al/eroeffng.htm, referenziert: 19.07.2003, III

[Promise, 2003] Promise Technology, Inc.: Support, *Webpage:* http://www.promise.com/support/support_eng.asp, referenziert: 19.07.2003, III

[Rathje, 2002] Rathje, U.: Technisches Konzept für die Datenarchivierung im Bundesarchiv, in: Der Archivar, Mitteilungsblatt für deutsches Archivwesen, Ausgabe 05/2002, Heft 2, Düsseldorf, Nordrhein-Westfälischen Hauptstaatsarchiv, 2002, ISSN: 0003-9500

[Rechtemanagement, 2001] Kompetenznetzwerk Universitätsverbund Multimedia NRW: Recht einfach - Rechtemanagement in Multimediaprojekten für Beteiligte am Bundesprogramm "Neue Medien in der Bildung", Themenforum Multimedia+Recht, Dortmund, 26.11.01 - 27.11.01, *Website:* http://www.uvm.nrw.de, referenziert: 19.07.2003, III

[Reinmann-Rothmeier, 2003] Reinmann-Rothmeier, G.: Didaktische Innovation durch Blended Learning, Leitlinien anhand eines Beispiels aus der Hochschule, 1. Auflage, Bern, Schweiz, Huber Verlag, 2003, ISBN: 3-356-83952-9, S. 28

[Reuter, 2002] Reuter, M./ Zacher, S.: Regelungstechnik für Ingenieure, Analyse, Simulation und Entwurf von Regelkreisen, 10. Auflage, Braunschweig, Vieweg Verlag, 2002, ISBN: 3-5289-4004-2

[Richtlinien, 1981] Technik, Sekundarstufe II, Gymnasium/Gesamtschule, Richtlinien und Lehrpläne, NRW, Nr. 4726, Frechen, Ritterbach Verlag, 1981

[Richtlinien, 1986] Technik, Realschule, Richtlinien und Lehrpläne, NRW, Nr. 3317, Frechen, Ritterbach Verlag, 1986, ISBN: 3-89314-255-x

[Richtlinien, 1989] Technik, Lernbereich Arbeitslehre, Hauptschule, Richtlinien und Lehrpläne, NRW, Nr. 3206/1, Frechen, Ritterbach Verlag, 1989, ISBN: 3-89314-077-8

[Richtlinien, 1993] Technik, Gymnasium, Sekundarstufe I, Richtlinien und Lehrpläne, NRW, Nr. 3422, Frechen, Ritterbach Verlag, 1993, ISBN: 3-89314-287-8

[Richtlinien, 1994] Leistungskurse im Fach Technik, Gymnasiale Oberstufe, Gymnasium/Gesamtschule, Leistungskurs, Richtlinien und Lehrpläne, NRW, Nr. 4731, Frechen, Ritterbach Verlag, 1994, ISBN: 3-89314-338-6

[Richtlinien, 1998] Technik, Arbeitslehre, Gesamtschule, Sekundarstufe I, Richtlinien und Lehrpläne, NRW, Nr. 3103, Frechen, Ritterbach Verlag, 1998, ISBN: 3-89314-557-5

[Richtlinien, 1999] Technik, Sekundarstufe II, Gymnasium/Gesamtschule, Richtlinien und Lehrpläne, NRW, Nr. 4726, Frechen, Ritterbach Verlag, 1999, ISBN: 3-89314-629-6

[Richtlinien, 1999a] Technik, Sekundarstufe II, Gymnasium/Gesamtschule, Richtlinien und Lehrpläne, NRW, Nr. 4726, Frechen, Ritterbach Verlag, 1999, ISBN: 3-89314-629-6, S. 16ff

[Richtlinien, 1999b] Technik, Sekundarstufe II, Gymnasium/Gesamtschule, Richtlinien und Lehrpläne, NRW, Nr. 4726, Frechen, Ritterbach Verlag, 1999, ISBN: 3-89314-629-6, S. 5

[Richtlinien, 1999c] Technik, Sekundarstufe II, Gymnasium/Gesamtschule, Richtlinien und Lehrpläne, NRW, Nr. 4726, Frechen, Ritterbach Verlag, 1999, ISBN: 3-89314-629-6, S. 6ff

[Ropohl, 1975] Ropohl, G. (Hrsg.): Systemtechnik – Grundlagen und Anwendung, München, Hanser Verlag, 1975, ISBN: 3-446-11829-2

[Ropohl, 1999] Ropohl, G.: Allgemeine Technologie, Eine Systemtheorie der Technik, 2. Auflage, München, Hanser Verlag, 1999, ISBN: 3-446-19606-4, S. 130-134

[Sachsen-Anhalt, 2001] Wirtschaftliche Bildung an allgemein bildenden Schulen, Bericht der Kultusministerkonferenz vom 19.10.2001, *pdf-Dokument:* http://www.kmk.org/doc/publ/wirt-bildung.pdf, referenziert: 05.02.2003, III

[SAPDB, 2003] PHP für Profis: SAPDB goes MySQL, *Webpage:* http://www.php-center.de/beitraege/detail.php?a_id=359, referenziert: 19.07.2003, III

[Sauer, 2002] Sauer, E./ Haupt, W.: Universitäre Lehrerausbildung für Technik-Unterricht in der Sekundarstufe II von Gymnasien, in: Technische Bildung, Ansätze und Perspektiven, 1. Auflage, Frankfurt am Main, Lang Verlag, 2002, ISBN 3-631-38548-X, S. 247-260

[Schmayl, 1995] Schmayl, W./ Wilkening, F.: Technikunterricht, 2. Auflage, Bad Heilbrunn, Klinkhardt Verlag, 1995, ISBN: 3-7815-0810-2

[Schmidt, 2002] Schmidt, J.: Nicht 'trustworthy', Internet Explorer gefährdet Rechner und Netze, in: c't, Magazin für Computertechnik, Ausgabe 25/2002, Hannover, Heise Verlag, 2002, ISSN: 0724-8679

[Schmitz, 2003] Schmitz, C./ Goldfuss, S./ Cielen, P.: ColdFusion MX, Professionelle Anwendungsentwicklung fürs Web, München Addison-Wesley Verlag, 2003, ISBN: 3-8273-2068-2

[Schneller, 2003] Schneller, A.: E-Learning tritt auf der Stelle, in: c't, Magazin für Computertechnik, Ausgabe 05/2003, Hannover, Heise Verlag, 2003, ISSN: 0724-8679

[Schulmeister, 2001] Schulmeister, R.: Virtuelle Universität Virtuelles Lernen, München, Oldenbourg Wissenschaftsverlag GmbH, 2001, ISBN: 3-486-25742-0

[Schulz, 2003] Schulz, H./ Siering, P.: Datendiener, Freie Datenbankserver im Vergleich, in: c't, Magazin für Computertechnik, Ausgabe 05/2003, Hannover, Heise Verlag, 2003, ISSN: 0724-8679

[Siebert, 2003] Siebert, H.: Pädagogischer Konstruktivismus, 2. vollständig überarbeitete und erweiterte Auflage, Unterschleißheim/München, Luchterhand, 2003, ISBN: 3-472-05317-8

[Spitzer, 2000] Spitzer, M.: Geist im Netz, Modelle für Lernen Denken und Handeln, Heidelberg, Spektrum Akademischer Verlag, 2000, ISBN: 3-8274-0572-6

[Stein, 1998] Stein, L.: Official Guide to Programming with CGI.pm, The Standard for Building Web Scripts, New York, John Wiley & Sons, Inc., 1998, ISBN: 0-471-24744-8

[StO, 1988] Studienordnung für das Unterrichtsfach Technik, Universität GH Essen, Bereinigte Sammlung der Satzungen und Ordnungen, Ziffer 9.37.1, 10. März 1988

[StO, 2000] Studienordnung für das Unterrichtsfach Technik, Fassung vom 31.03.2000, Archiv des Faches Technologie und Didaktik der Technik TUD

[TDT, 2001] Technologie und Didaktik der Technik, TdT, *Website:* http://technologie.uni-duisburg.de/tdt/home.htm, referenziert: 19.07.2003, III

[TeachersNews, 2003] TeachersNews, Die aktuelle Informationsquelle rund um das Thema Schule, *Website:* http://www.teachersnews.net/index.php?shownews=906, referenziert: 12.07.2003, III

[Technikdidaktik, 2003] Technikdidaktik und Umwelterziehung, Martin-Luther-Universität, Ingenieurwissenschaftliche Fakultät, Halle-Wittenberg, *Webpage:* http://www.iw.uni-halle.de/awtb/haupt.html referenziert: 19.07.2003, III

[TeLos, 2003] Technikspezifische Lernobjekte, *Website:* http://lpf.tud.uni-essen.de/homepages/TeLos/IndexTeLos.htm, referenziert: 19.07.2003, III

[TemplatesDC, 2003] Template-Generator nach Dublin Core, MultiMediaDatenBank TechnikUnterricht, *Webpage:* http://fred2.technik-didaktik.uni-essen.de/~mmdb-tu/mmdb/dc_template.htm, referenziert: 19.07.2003, III

[TERC, 2002] Wehling, J./ Haupt, W.: Multimedia in Technology Teacher Education, in: Learning in Technology Education: Challenges for the 21st century, Volume 2, Gold Coast, Queensland, Australia, Centre for Technology Education Research, Griffith University, 2002, ISBN: 0-909291-74-8

[Throll, 2002] Throll, M.: MySQL, Grundlagen, Entwicklung, Administration, 1. Auflage, Bonn, Galileo Press Verlag, 2002, ISBN: 3-89842-188-0

[TUD, 2003] Technologie und Didaktik der Technik, *Website:* http://www.tud.uni-essen.de, referenziert: 19.07.2003, III

[TUF, 2003] Technik Unterricht Forum e.V., *Website:* http://www.tuf-ev.de, referenziert: 19.07.2003, III

[Unicode, 2003] Unicode, Inc., Website: http://www.unicode.org, referenziert: 19.07.2003, III

[Unix, 1999] Unix Configuration Guidelines, CERT Coordination Center, *Webpage:* http://www.cert.org/tech_tips/unix_configuration_guidelines.html, referenziert: 19.07.2003, III

[UVM, 2003] Kompetenznetzwerk Universitätsverbund MultiMedia NRW, *Website:* http://www.uvm.nrw.de, referenziert: 19.07.3003, III

[UVM, 2003a] Kompetenznetzwerk Universitätsverbund MultiMedia NRW, Projekte, *Webpage:* http://www.uvm.nrw.de/projekte/, referenziert: 19.07.3003, III

[UVM, 2003b] Kompetenznetzwerk Universitätsverbund MultiMedia NRW, Multimediarecht für die Hochschulpraxis, *Website:* http://www.uvm.nrw.de, referenziert: 19.07.03, III

[VDI, 2003] Verein Deutscher Ingenieure, *Website:* http://www.vdi.de, referenziert: 19.07.2001, III

[VDI Nachrichten, 2003] Anne Schneller, Im Netz allein liegt nicht die Zukunft, VDI Nachrichten vom 17.01.2003, *Webpage:* http://www.vdi-nachrichten.com/vdi_nachrichten/aktuelle_ausgabe/akt_ausg_detail.asp?ID=10810, referenziert: 08.02.2003, III

[VdS, 2003] Verband der deutschen Schulbuchverleger: Schulbücher werden 9 Jahre alt, in: Süddeutsche Zeitung, Nr. 73, 28.03.2003, Seite 7

[VIB, 1999] Virtualisierung im Bildungsbereich, Verbundprojekt der Pädagogischen Hochschulen des Landes Baden-Württemberg, Didaktische Modelle, *Webpage:* http://www.vib-bw.de/tp4/Start/regal/ordner/markus/, referenziert: 19.07.2003, III

[W3C, 2003] WorldWideWeb Consortium. *Website:* http://www.w3.org, referenziert: 19.07.2003, III

[W3C, 2003a] WorldWideWeb Consortium: Ressource Description Framework (RDF). *Webpage:* http://www.w3.org/RDF/, referenziert: 19.07.2003, III

[W3C, 2003b] WorldWideWeb Consortium: HyperText Markup Language (HTML), *Webpage:* http://www.w3.org/MarkUp/, referenziert: 19.07.2003, III

[W3C, 2003c] WorldWideWeb Consortium: Extensible Markup Language (XML), *Webpage:* http://www.w3.org/XML/, referenziert: 19.07.2003, III

[W3C, 2003d] WorldWideWeb Consortium: The Extensible Stylesheet Language (XSL), *Webpage:* http://www.w3.org/Style/XSL/, referenziert: 19.07.2003, III

[W3C, 2003e] WorldWideWeb Consortium: XHTML 2.0, W3C Working Draft 31 January 2003, *Webpage:* http://www.w3.org/TR/2003/WD-xhtml2-20030131/, referenziert: 19.07.2003, III

[W3C-DTF, 1997] WorldWideWeb Consortium, Date and Time Formats, *Webpage:* http://www.w3.org/TR/NOTE-datetime, referenziert: 19.07.2003, III

[Webalizer, 2003] The Webalizer, What is your web server doing today, *Webpage:* http://www.mrunix.net/webalizer/, referenziert: 19.07.2003, III

[WebCT, 2003] WebCT Inc., the world's leading provider of e-Learning solutions for higher education, *Website:* http://www.webct.com, referenziert: 19.07.2003, III

[Wehling, 1996] Wehling, J.: Skriptum zur Fachdidaktik der Maschinen- und Fertigungstechnik, Universität-GH-Essen, Fachbereich 12, Technologie und Didaktik der Technik TUD, 1996, Kap. 4, S. 11

[Wehling, 2002] Wehling, J.: Lehrerausbildung für das Fach Technik im Gymnasium im Bereich "Informationstechnologie", in: Technische Bildung, Ansätze und Perspektiven, 1. Auflage, Frankfurt am Main, Lang Verlag, 2002, ISBN 3-631-38548-X, S. 261-266

[Wehling, 2003] Wehling, J.: Datenverarbeitung III, Datenverarbeitung IV, *Webpage:* http://www.tud.uni-essen.de/wehling/dat3.htm, referenziert: 19.07.2003, III

[Wehling, 2003a] Wehling, J.: Einrichtung eines Webservers und Datenbankservers unter SuSE Linux 8.1 Professional, *Webpage:* http://www.mmdb-tu.de/servereinrichtung/lamp.htm, referenziert: 02.04.2003, III

[Weichselbraun, 2002] Weichselbraun, A.: Metadaten im Web, Diplomarbeit, 2002, *pdf-Dokument:* http://wwwai.wu-wien.ac.at/usr/albert/, referenziert: 19.07.2003, III

[Weinert, 1996] Weinert, F. E. (Hrsg.): Enzyklopädie der Psychologie, Band 2, Psychologie des Lernens und der Instruktion, Göttingen, Hogrefe Verlag, 1996, ISBN: 3-8017-0538-2

[Weinert, 1997] Weinert, F. E. (Hrsg.): Enzyklopädie der Psychologie, Band 3, Psychologie des Unterrichts und der Schule, Göttingen, Hogrefe Verlag, 1997, ISBN: 3-8017-0539-0

[Wiki, 2002] What is Wiki? Website: http://wiki.org/wiki.cgi?WhatIsWiki, referenziert: 19.07.2003, III

[Wilde, 2003] Wilde, E.: Das bessere Schema, XML 2002: Datenbanken im Zentrum, in: iX, Magazin für professionelle Informationstechnik, Ausgabe 02/2003, Hannover, Heise Verlag, 2003, ISSN: 0935-9680

[Wilkening, 1996] Wilkening, F./ Lindemann, K./ Schmayl, W.: Unterrichtsverfahren im Lernbereich Arbeit und Technik, Ravensburg, Ravensburg Verlag, 1996, ISBN: 3-4736-1643-5

[Zentel, 2002] Zentel, P./ Bett, K./ Meister, D./ Rinn, U./ Wedekind, J.: Trends und Perspektiven der virtuellen Hochschule in Deutschland, in: it - Information Technology, Methoden und innovative Anwendungen der Informatik und Informationstechnik, Ausgabe 4/2002, München, Oldenbourg Verlag, 2002, ISSN: 0944-2774